(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
8 February 2001 (08.02.2001)

PCT

(10) International Publication Number
WO 01/09803 A1

(51) International Patent Classification[7]: G06F 17/60, 153/00, 19/00

(21) International Application Number: PCT/US00/21281

(22) International Filing Date: 3 August 2000 (03.08.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/147,032 3 August 1999 (03.08.1999) US

(71) Applicant *(for all designated States except US)*: **BIOS GROUP INC.** [US/US]; 317 Paseo de Peralta, Santa Fe, NM 87501 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **KAUFFMAN, Stuart, A.** [US/US]; 1811 South Camino Cruz Blanco, Santa Fe, NM 87505 (US). **HERRIOT, James, W.** [US/US]; 784 Rosewood Drive, Palo Alto, CA 94303 (US). **YAKUSHIN, Igor** [BG/US]; 1207 Agua Fria Street, Santa Fe, NM 87501 (US). **VALDIMAR, Makhanov** [RU/US]; 4148 Chatarron Place, Santa Fe, NM 87505 (US). **PLATE, Tony** [US/US]; 812 Loma Boreal, Santa Fe, NM 87501 (US). **TITELBAUM, Dan** [US/US]; 618 Old Santa Fe Trail, #G, Santa Fe, NM 87501 (US).

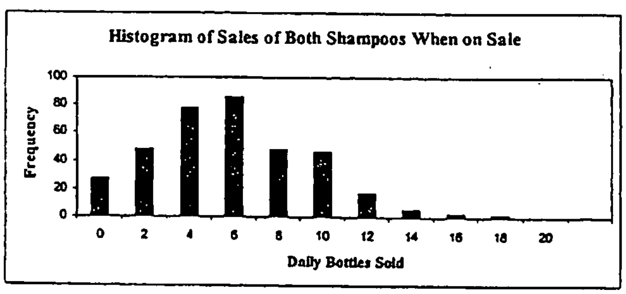
(74) Agents: **MORRIS, Francis, E.** et al.; Pennie & Edmonds LLP, 1155 Avenue of the Americas, New York, NY 10036 (US).

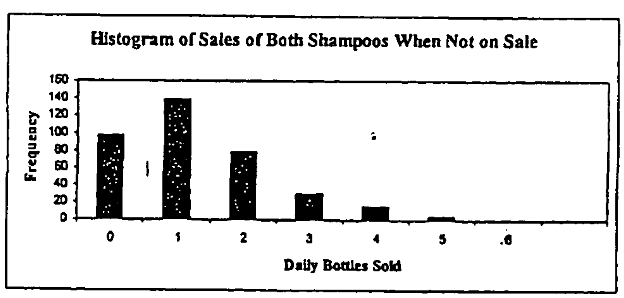
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: A METHOD AND SYSTEM FOR SUPPLY CHAIN MANAGEMENT





(57) Abstract: The present invention relates generally to a method and system for supply chain management. More particularly, the present invention searches for supply chain parameter values and supply chain policies that are optimal with respect to a group of performance measures including inventory, time of inventory, the number of short shipments and cost.

WO 01/09803 A1

IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

A METHOD AND SYSTEM FOR SUPPLY CHAIN MANAGEMENT

FIELD OF THE INVENTION

The present invention relates generally to a method and system for supply chain management. More particularly, the present invention searches for supply chain parameter values and supply chain policies that are optimal with respect to a group of performance measures including inventory, time of inventory, the number of short shipments and cost.

Background

Supply chains involve multiple companies, working both competitively and cooperatively, to produce, transport, and sell products to the consumer. Supply chains span from initial suppliers of raw materials, through transport and assembly of raw materials, often in warehouses or holding tanks, into manufacturing or assembly centers that combine two or more raw materials via processes that lead to intermediate or final products, to the storage of such products on pallets or other devices, the warehousing of such products, their transport to intermediate distribution centers, their transportation from such centers to other distribution centers and thence to retail outlets or the final consumer.

In general, the dynamics of such a supply chain are highly complex. Among the factors entering the behavior of the supply chain are lead times and order cycle time, which govern the step up in inventory at down stream holding which is thereafter drawn down, the multiplicity of each input and the delays associated with manufacture at each step, and transport at each step, the topology of the supply chain as a flow network, (often a star, or hub and spoke configuration),

the uncertainty in demand from the final consumer either in the absence or presence of periodic promotional sales of one or more of the Stock Keeping Units (SKUs) available.

Supply chains must also operate within an environment that has market failures. In 1776, Adam Smith proclaimed that an invisible hand guides individual agents, each making self decisions, towards globally efficient solutions. We have learned since then that this is usually, but not always, the case because of market failures. Market failures include lack of information, externalities, lack of a market, public goods, transportation costs and monopolies. Market failures can guide the economy to places where selfish decisions lead to inefficient outcomes. Governments fight market failures using administrative agencies such as the FDA and health inspection agencies, pollution laws, insurance companies, taxes for highways and national defense, and laws governing monopolies.

In conventional supply chain management, each company has optimized its own piece of the supply chain, ignoring any detrimental effects that its decisions may have on its supply chain partners. But companies are now realizing that local, selfish optimization has reached its limits.

Accordingly, there exists a need for a tool that can examine the global impacts of local decisions. With such a tool, supply chain partners could examine the implications and effects of their actions and make choices which lead to superior outcomes.

Summary of the Invention

The present invention is a method and system to manage a supply chain by examining the global impact of local

decisions. Using the present invention, supply chain partners can examine the implications and effects of their actions and make choices which lead to superior outcomes.

The present invention includes a method for managing at least one supply chain comprising the steps of: defining at least one model having a parameter space to represent the supply chain; defining a performance space; defining at least one policy for the supply chain; and softening said at least one policy.

The present invention further includes a method for managing a supply chain involving one or more sites comprising the steps of: defining one or more models for the one or more sites having one or more parameters, said one or more parameters having one or more settings; defining a performance space; and searching for one or more optimal values of said settings of said one or more parameters comprising the steps of: changing at least one of said settings at one or more of the sites; and simulating said one or more models to generate one or more corresponding values in said performance space.

Brief Description of Drawings

FIG. 1 shows a histogram of sales of shampoos when they are on sale and a histogram of sales of shampoos when they are not on sale.

FIG. 2 shows data for stores that choose sales randomly.

FIG. 3 shows shelf inventory data for random sales with no information sharing.

FIG. 4 shows Distribution Center inventory data for random sales with no information sharing.

FIG. 5 shows daily sales at one store of two different products.

FIG. 6 shows shelf inventory data for random sales with information sharing.

FIG. 7 shows Distribution Center inventory data for random sales with information sharing.

FIG. 8 shows actual point of sale data.

FIG. 9 shows daily sales data.

FIG. 10 shows shelf inventory data with no information sharing.

FIG. 11 shows retail Distribution Center data with no information sharing.

FIG. 12 shows simulated data for daily sales.

FIG. 13 shows store shelf data with information sharing.

FIG. 14 shows retail Distribution Center data with information sharing.

FIG. 15 shows total sales, out of stock and inventory data for information sharing, no information sharing, big truck sizes, and little truck sizes.

FIG. 16 shows annual earnings data at different sites in the supply chain when stores employ different numbers of stock boys.

FIG. 17 illustrates the building blocks of the hydrodynamic model of a supply chain.

FIG. 18 illustrates the representation of the left boundary condition and the right boundary condition.

FIG. 19 illustrates an analogous hydrodynamic flow through the model.

FIG. 20 illustrates equivalence schemes of the hydrodynamic model.

FIG. 21 illustrates the generic plant by a production line, an inventory which stores raw materials, and an inventory which stores the final product.

FIG. 22 illustrates a supply chain model and its equivalent scheme.

FIG. 23 shows the simulation results for the supply chain model and its equivalent from FIG. 22.

FIG. 24 illustrates bottlenecks: local vs. global optimization.

FIG. 25 shows a mapping from a parameter space to a performance space.

FIGs. 26 and 27 show several hundred points uniformly distributed in the parameter space that are mapped into the performance space.

FIG. 28 corresponds to the perfect world case and rigid policies.

FIG. 29 shows how small noise in multiplicity destroys the optimal performance point.

FIG. 30 shows the restoration of the optimal performance point when the policies are softened.

FIG. 31 shows that softening the policy in the perfect world widens the range of optimal parameters.

FIG. 32 shows the out-of-stocks for a pool of 5 stores with standard allocation and with reallocation.

FIG. 33 shows out-of-stocks for a pool of 5 stores with standard allocation and with reallocation.

FIG. 34 shows excess inventory for a pool of 5 stores with standard allocation and with reallocation.

FIG. 35 shows the time-in-system for a pool of 5 stores with standard allocation and with reallocation.

FIG. 36 shows out-of-stocks for standard allocation and reallocation in a log-log plot.

FIG. 37 shows out-of-stocks for standard allocation and for reallocation for 3, 5, and 10 stores.

FIG. 38 shows excess inventory for standard allocation and for reallocation for 3, 5, and 10 stores.

FIG. 39 shows the time-in-system for standard allocation and reallocation for 3, 5, and 10 stores.

FIG. 40 shows a landscape of trigger level to profit for the reallocation model.

FIGs. 41 and 42 show landscapes of batch size to total profit.

FIG. 43 shows an example progress of a tau-search when tau is one and maxjump is six.

FIG. 44 shows an example progress of a tau-search when tau is five and maxjump is six.

FIG. 45 shows an example progress for a tau-search when tau is one and maxjump is one.

FIG. 46 shows the fitness of the best configuration that was found.

FIG. 47 shows a chart of the steps that were accepted in the search.

FIG. 48 shows the tries to reach the best configuration that was found.

Detailed Description of the Preferred Embodiment

The present invention searches for supply chain parameter values and supply chain policies that are optimal with respect to a group of performance measures including inventory, time of inventory, the number of short shipments and cost.

Supply Chain Model

The present invention includes a model of a supply chain having five agents: consumers, stores, Distribution Centers (DCs), transport and factories. The model constructs itself from data files including delivery files and shipment files. The models can be run in at least two modes: 1) calibrate/push or 2) pull. In the calibrate/push mode, the shipment data drives the model while in the pull mode, the consumer demand drives the model.

Each consumer agent has between 0 and 1 (preferably uniformly distributed) bottle of shampoo in the house. In one embodiment, she washes her hair every morning consuming between 1/75th and 2/25ths of the bottle (preferably normally distributed). When she finishes the bottle, she goes to the store to buy another.

Consumer agents have at least two levels of demand: 1) high if the product is on sale and 1) low, otherwise. Consumers prefer to buy on-sale products, but have a small chance of buying non-sale products. Preferably, the chance of buying on sale and non-sale products can be varied by the modeler. Consumers can choose to purchase up to five times as many bottles of on-sale shampoo. Preferably, the difference in volume purchased can be varied by the modeler.

When a product is out of stock at the store, the consumer leaves. If this happens, she may then choose never to buy that brand of shampoo again. Preferably, the probabilities of these occurrences (brand loyalty) are controlled by the modeler.

If the product is in stock, the consumer may buy goods other than just shampoo. This parameter, called impulse-buy is preferably controlled by the modeler. An exemplary value for this parameter is 1-5 times the shampoo price.

The method and system of the present invention further includes a calibration utility which varies consumer demand parameters to meet actual supply chain data including deliveries from a production facility to distribution centers and out of stock at shelf data, which can be obtained through interviews. FIG. 1 shows a histogram of sales of shampoos when they are on sale and a histogram of sales of shampoos when they are not on sale.

The stores in the model of the present invention set inventory levels using different techniques. Exemplary techniques for setting inventory levels includes predicting consumer demand based on past history and upcoming sales events, estimating a 99% confidence interval and ordering products from retail Distribution Centers.

The stores choose when to have sales and choose the number of stock boys to employ. Each night, the store receives delivery trucks. Trucks are unloaded and product is stored in the backroom. Stock boys then move product from the back room to the store shelf. Product can get lost in the backroom, however. Total stock lost in the backroom decreases with number of stock boys and increases with total inventory stored in the backroom.

Stock boys affect shelf volumes which in turn affects total sales and POS which in turn causes defection of customers to other stores and other products. The costs of employing stock boys are fully borne by the store but the benefits can be felt throughout the chain. This model allows quantifies those supply chain-wide benefits.

Distribution Centers set inventory levels using different techniques. Exemplary techniques include predicting store demand based on past history and upcoming sales events, estimating a 99% confidence interval, and ordering products from factories.

Factories set inventory levels using different techniques. Exemplary techniques include predicting Distribution Center demand based on past history, estimating a 99% confidence interval, ordering raw ingredients from supplier factories and producing finished goods from raw ingredients.

Experiments were run with the supply chain model of the present invention to show the influence of information sharing on the performance of the supply chain. FIGs. 2-13 show sales data and performance data for this experiment.

In particular, FIG. 2 shows data for stores that choose sales randomly. FIG. 3 shows shelf inventory data for random sales with no information sharing. FIG. 4 shows Distribution Center inventory data for random sales with no information sharing. FIG. 5 shows daily sales at one store of two different products. FIG. 6 shows shelf inventory data for random sales with information sharing. FIG. 7 shows Distribution Center inventory data for random sales with information sharing. FIG. 8 shows actual point of sale data. FIG. 9 shows daily sales data. FIG. 10 shows shelf inventory data with no information sharing. FIG. 11 shows retail

Distribution Center data with no information sharing. FIG. 12 shows simulated data for daily sales. FIG. 13 shows store shelf data with information sharing. FIG. 14 shows retail Distribution Center data with information sharing.

As indicated by these figures, the model of the present invention demonstrates that information sharing can increase sales, reduce out of stocks and greatly reduce system inventory.

Additional experiments were run with the model of the present invention to show the effect of truck size on the performance of the supply chain. FIG. 15 shows total sales, out of stock and inventory data for information sharing, no information sharing, big truck sizes, and little truck sizes.

Additional experiments were run with the model of the present invention to show the effect of employing stock boys. FIG. 16 shows annual earnings data at different sites in the supply chain when stores employ different numbers of stock boys. This experiment illustrates a positive externality. In particular, the cost of employing stock boys is completely borne by the stores but the positive effects of stock boys accrue to all members of the supply chain and the consumers. In other words, store efficiency at moving products from the backroom to the shelf has repercussions throughout the supply chain. Thus, what is locally optimal for the store may not be globally optimal for the entire chain. For example, a company might reduce out-of-stocks on its products if it packed them in boxes painted safety orange, or some other easily distinguishable color.

Hydrodynamic Supply Chain Model

The present invention further includes a hydrodynamic model of a supply chain with a number of capabilities. For example, the model can facilitate the study of the optimal mode of operation for a supply chain when the flow of Stock Keeping Units (SKUs) is limited only by the physical properties of the chain and not by consumer demand. The model also facilitates the study of the sensitivity of the optimal regime to noise.

FIG. 17 illustrates the building blocks of the hydrodynamic model of a supply chain. The model includes an inventory or warehouse having a capacity and a counter. The model further includes a link having a multiplicity which includes shipment size and batch size. The model further includes a production line (factory or transportation) having a delay.

In this model, the supply chain is represented by a graph where inventories can be connected to production lines and production lines can be connected to inventories. Preferably, two inventory nodes are never connected directly to each other and two production lines are never connected to each other. Exemplary graphs include a bipartite graph where inventories are connected via production lines. Preferably, each inventory stores one type of SKU. One can combine several inventories into one inventory that can store many types of SKU. The same is true for production lines.

At each time step, all the production lines are called in a random order. Each production line checks its current state. It might be time to request resources or to distribute product. It might be in the middle of a production cycle or it might have to wait for resources or wait to distribute its product.

The hydrodynamic model includes a number of policies. In one exemplary policy, a production line waits until all needed resources are available simultaneously from the upstream warehouses in the required amounts. Otherwise, the production line waits. In another exemplary policy, the production line tries to distribute all the produced SKUs to some downstream warehouse when production is finished. In this exemplary policy, the production line stops producing and waits if the downstream warehouse does not have enough space to accept all the SKUs.

This model is called hydrodynamic because the flow of SKUs through the supply chain happens in a hydrodynamical fashion. If raw materials are available, production lines get the ingredients and start producing. If there is available space int he downstream warehouses, the produced SKUs are moved forward down stream. Otherwise, the production line waits. SKUs flow through the supply chain like water flows through pipes - only limited by their capacity to carry water (multiplicity), the storage capacity of reservoirs downstream (inventory capacity) and, of course, the supply of water upstream (high initial counter at inventories with raw materials). Within the framework of this analogy, introducing orders corresponds to introducing switches into the system that would turn the flow on and off.

The hydrodynamic model represents boundary conditions. Any left boundary condition (supply side) can be modeled by an inventory with infinite supply of SKUs and a production line with specified delay and multiplicity of the outgoing links. Any right boundary conditions (demand side) can be modeled by the opposite construction: a production line and the black hole (an inventory that can accept any number of SKUs). By changing the multiplicity of the

incoming link and delay of the production line, the model can simulate any demand curve. FIG. 18 illustrates the representation of the left boundary condition and the right boundary condition.

FIG. 19 illustrates an analogous hydrodynamic flow through the model. Since the box in the middle denotes a subchain, the whole process can be regarded as a flow from the left reservoir to the right one. Suppose $m_L = m_R = m$ and $d_L = d_R = d$ and there are no sources or sinks inside the box. Then we have something analogous to a hydrodynamic flow through the system with the total current defined as $J_{tot} = m/d$. This resembles an electrical circuit with the multiplicity being an electric charge.

FIG. 20 illustrates equivalence schemes of the hydrodynamic model. Schemes with different topologies are considered equivalent in the sense that for the same input they would produce the same output. The topologies to the right of the arrows in FIG. 20 are equivalent to the topologies to the left of the arrows under the conditions shown on the right. In the second picture, one must also add inventory levels in the 2nd and 3rd warehouses on the left hand side and equate the sum to the inventory level of the warehouse 2_3 on the right hand side. Using the equivalence relations, one can reduce a supply chain of complicated topology with a large number of nodes to a simple linear generic supply chain which consists of left and right boundary constructions and a generic plant in the middle. The generic plant is modeled by a production line, an inventory which stores raw materials, and an inventory which stores the final product as illustrated in FIG. 21.

These equivalence schemes are very important because they improve the computational performance of the

model. In particular, if one wants to optimize a small part of the supply chain, one can replace the rest of the chain with a small equivalent scheme to significantly speed up the computation.

FIG. 22 illustrates a supply chain model and its equivalent scheme. FIG. 23 shows the simulation results for the supply chain model and its equivalent from FIG. 22. The results show that the equivalent scheme is a very close representation of the original supply chain.

FIG. 24 illustrates bottlenecks: local vs. global optimization. If one considers the plant and customer parts of the generic supply chain and assumes that the multiplicity is the same along this piece of the supply chain, then the bottleneck is defined by the longest delay in production lines PL 2 and PL 3. If $d_2 > d_3$, the bottleneck is at the plant: it operates up to its capacity and its inventory does not grow. The customer, on the other hand, suffers from short shipments. This is the pull regime.

If $d_2 < d_3$, the bottleneck is at the customer: it consumes up to its capacity and does not experience any short shipments. The producer, on the other hand, suffers from growing inventory. This is the push regime.

The global optimal behavior of this piece of the supply chain takes place when $d_2 \approx d_3$. Experiments indicate that this resonance case is the most vulnerable to noise.

Next, we provide some justification for considering the hydrodynamic model instead of a full blown model of the supply chain with all the details. The model admits a laminar hydrodynamic flow of materials (SKU's) through the chain which by definition is the most optimal flow under specified parameters in the perfect world (adding orders to the model can only slow its performance down by imposing

constraints, that is why the first step to study a supply chain is to consider hydrodynamic model which is only constrained by physical parameters of the chain: sizes of the warehouses, multiplicities, time delays in production lines, and topology). The model has the parameter space and the performance space. The production cycle constitutes a mapping from the first into the second. The performance space coordinates are the characteristics to be optimized, such as the total inventory in a piece of a chain, short shipments, time needed for an SKU to go through that piece, average level of inventor of the final product at the plant C, an average number of short shipments at the same inventory SS, an average amount of time needed to produce on SKU T. The parameters are those that have an effect on the production cycle. Exemplary parameters include time delays in the production lines, multiplicities M (intensity of the flow) at transportation production line R and warehouse capacities. FIG. 25 shows a mapping from a parameter space to a performance space. To map a point from the parameter space to performance space, one can choose a point in the parameter space run the simulator for N time steps, average C, SS and T over the last $n < N$ steps.

In the most optimal regime, a point in the parameter space is tuned to produce a laminar flow through the chain piece under consideration: in the absence of noise (perfect world), there can be zero shorts, zero inventories and minimal time to produce on SKU. Uncertainties of the real world naturally destroy this ideal picture (a parameter resonance). The model helps to determine which parameters of the supply chain are the most sensitive, how the optimal regime breaks, what is the most optimal regime in the noisy work that can be reached and how to reach the optimal regime.

Preferably, the model comprises algorithms to optimize policies at each node. Preferably, the model is order driven.

The model can analyze the relationship between total inventory in the system, high versus low, and the behavior of the system in either a classical or quantum regime. Here, the issue is related to what we are calling "integer" problems. Consider as a non-limiting example, that the multiplicity of units used in manufacture are identical to the multiplicity of units transported in from a supplier, say 10 and 10. This is a match condition. Conversely, the transport quantity from the supplier may be 17, while the use of the material in each round of manufacture may be 11. These non-matches, or "integer" lumpiness mismatches, tend to create the need to hold and transport extra inventory, leads to extra time in the system, and tends to lead to short shipments. This "integer" lumpiness results in rugged, multi-peaked, "fitness landscapes" for each of the figures of merit. In general, if the system is operating with abundant inventory, in the "classical regime", then the lumpiness does not matter so much. However, as inventory is removed from the supply chain, the system enters the "quantum regime" where endogenous noise arises due to the integer lumpy character of the supply chain. Thus, in general, as the supply chain attempts to operate with decreasing inventory, the ruggedness of the landscape increases. In turn, this leads to two requirements: a) A need for decision support tools to find good operating regimes in the rugged landscape, including optimization of search. b). The need to soften the integer constraints in many possible ways, in particular via automated markets and options to hedge risk, such that the real time value of softened constraints can be valued. For

example, it may dramatically soften constraints to allow less than full truck load deliveries, and introduce flexibility in the time of delivery to a distribution center or retail outlet, or to a final consumer. Proper real time pricing of the value of that policy, given costs of set up and holding and transport contrasted to overall benefits to the flow of the supply chain requires new markets, including pricing, and options.

Our analysis of different modes of operation of the supply chain demonstrates that optimal performance occurs in a regime of "laminar flow", where input multiplicity of each unit used in assembly or transport, and the delays of each step in the process, are identically matched. In this regime, all three figures of merit can he optimized simultaneously: The system can operate with virtually no inventory in the system, with no short deliveries, and with minimal time of each unit in the system. This laminar flow circumstance corresponds roughly to "lean manufacturing". However, this regime is unstable. Notably, if very small fluctuations are present in the "right hand boundary" representing consumer demand fluctuations, then this optimal behavior rapidly degrades, and the system must operate with substantial inventory in order to hold shorts to an acceptable minimum arid lime of SKUs in the system to a low level. This strong result demands that the supply chain be arranged such that it smooths out, and reacts readily to, fluctuations in consumer demand. Such operations allow the supply chain to operate in the "quantum regime" with low inventory. In addition, we have shown that softening of integer constraints tends to "stabilize" the optimal laminar flow regime in the space of nearby modes of operating the supply chain. In other words, if the integer constraints are

"hard edged", then slight deviations from perfect integer matching lead to bad system performance. With softening, for example, by allowing less than full truck loads, and flexibility in time of delivery, the laminar flow regime is broadened and the fitness landscape is smoothed. Thus, softening integer constraints is important to stabilize optimal performance regimes. In general, if we compare optimal performance, i.e. laminar flow and softening of integer constraints, with typical performance, our models suggest that optimal performance can yield a reduction in total inventory of threefold or more.

When some noise is added to the chosen point in the parameter space, the point is interpreted as an average value rather than a precise fixed number. FIGs. 26 and 27 show several hundred points uniformly distributed in the parameter space that are mapped into the performance space. We re interested in the points on the pareto optimal surface: minimum values for C, SS, T. We are also interested in how the map is affected by noise.

Results indicate that the map has a high complexity and is non-linear. There is a very good point in the perfect world which optimizes all performance characteristics but this point disappears as one increases noise in the system. When noise is added, isolated islands of attractors, which were present in the perfect world, spread out and glue with other islands. This looks very much like a phase transition.

Next, we consider the stability of the laminar flow in a supply chain. Here, we consider the most optimal region of the performance space and make inverse mapping back to the parameter space to study the stability problem. The latter reduces to the questions about the features of the mapping: whether it is a contraction, an expansion or an isometry. In

the first case one can say that the mapping is stable at least under small perturbations (noise). In the second case, it is unstable and even a small noise can destroy the "optimal" island. The third case needs a thorough analysis. Some conclusions from experiments with the model are as follows:

The inverse map of the optimal island gives for the perfect world case the tuned parameters corresponding to the laminar flow <$counter_I$> = 0, <$shortShipments_I$> = 0, and <time> = t_{min} for $d = d_{res}$, $m = m_{res}$.

The average inventory level is much more sensitive to the noise in multiplicity than to the noise in delay. As noise increases two new islands with different average inventory levels seem to appear. The first corresponds to the pulling regime with $d < d_{res}$ and relatively small <$counter_I$>, the second with $d >= d_{res}$ and relatively large <$counter_I$>, corresponds to the pushing regime.

The average number of short shipments is much more sensitive to the noise in delay than to the noise in multiplicity

In our experiments we chose the capacity of each inventory to be very large in comparison with average level. Under such conditions, time to produce one SKU is very weakly sensitive to the choice of parameters. However, if inventory capacities are small enough, time can increase to infinity for some choice of the parameters.

This system is very sensitive to noise. Further, close points in the parameter space might map to the points in the performance space which are very far apart from each other and vice versa. This is due to the rigid simplistic policies (all or nothing) implement in each node. To reduce the influence of noise and to increase stability of the system, one must use more flexible policies.

Softening policies make the optimal point more stable with respect to noise. We change a policy of the last production line. Previously, it would take strictly M SKUs rather than M-K. If the inventory upstream did not have at least M SKUs available, the production line had to wait. Now it will take M-K SKUs instead with the probability: $p=\exp(-K^2/2\ \text{sigma}^2)$. For example, suppose M=142 and sigma = 0.015* M. If only 141 SKUs are available, then $p = \exp(-(142-141)^2/2*(142*0.015)^2) = 0.9$. If only 140 SKUs are available, then p = 0.6.

FIGs. 28-31 show plots of average inventory level versus multiplicity for different delays. FIG. 28 corresponds to the perfect world case and rigid policies. FIG. 29 shows how small noise in multiplicity destroys the optimal performance point. FIG. 30 shows the restoration of the optimal performance point when the policies are softened. FIG. 31 shows that softening the policy in the perfect world widens the range of optimal parameters.

Accordingly, a supply chain of complicated topology can be reduced to a simpler chain using the equivalence rules if one is only interested in its averaged output performance. Policy softening leads to significant stabilization of the optimal point with respect to noise. Average inventory level

near the optimal point drops. Average number of short shipments remains about the same as with hard policy. Preferably, shortening the policy should be based on computing actual costs of relaxing constraints. Therefore, an automated market should be the mechanism that makes the laminar flow robust with respect to noise in the system. Both automated markets as well as the features affecting supply chain performance are described in international application number PCT/US99/15096 filed July 2, 1999 and titled, "An Adaptive and Reliable System and Method for Operations Management", the contents of which are herein incorporated by reference.

The model of the present invention further includes algorithm nodes. To have more sophisticated control of a supply chain operation and to learn an optimal behavior, algorithm nodes should be connected to a chain. An algorithm node is connected to three sets of nodes which might overlap: a set of nodes from which the algorithm learns about the state of a (sub)chain, a set of nodes from which an algorithm reads rewards for the previously taken actions, and a set of nodes on which an algorithm acts. It is the responsibility of each involved node to compute its state, reward and to tell the algorithm what is the possible choice of actions this node can take. The algorithm can implement some static if-then policy or it can learn the optimal strategy from experience using some reinforcement learning techniques.

In the real supply chain the same production line might serve to produce different types of SKUs using different recipes for production. Within the framework of our model we can represent such a situation as a combination of several production line nodes and an algorithm node. Algorithm node would ensure that only one of these production

lines is active at a time based on some optimization strategy.

Similar arrangement can be used to represent a warehouse that stores different types of SKUs. An algorithm node would ensure here that the total capacity of the warehouse is not exceeded and incompatible Sus are not stored together.

Simplified Order-Driven Model

The present invention further includes a simplified order-driven model of a supply chain. This model has the same building block as the hydrodynamical model. The order-driven model has a different flow of events. Active entities are inventories. Inventories can generate orders in several different ways. These orders would propagate all the way upstream if necessary during one time step. Production line would only request a shipment if it has a pending order for it. An order does not specify at which moment it should be fulfilled. It would be satisfied SAP. The hydrodynamical model is a limiting case of order-driven one when there is no shortage of orders in the chain.

An inventory node can read orders from a file. This is a way to enter the historical data into the model. An inventory node can generate orders based on three parameters: minimum, target, and frequency. After each frequency time steps an inventory checks its unit counter. If it is less than minimum, than target-counter units are ordered. An inventory node cen generate orders based on two parameters: frequency and size. After each frequency time steps an inventory orders size units.

When production line gets an order it rounds it up to the nearest integer divisible by the output multiplicity

and orders the needed amounts of ingredients. An inventory node has now three types of SKUs: SKUs packed into a shipment of the corresponding multiplicity and ready to be taken (readyShipments list); SKUs reserved for some previous orders but not ready to be shipped yet (readyOrders list); SKUs that are available and not reserved.

When an inventory gets an order, it checks if it can be satisfied from the pool of available units. If it can be satisfied completely, the corresponding amount of SKUs are moved to readyOrders list. If it can be satisfied partially, whatever is available is moved to readyOrders, the rest is ordered. If nothing is available, everything is ordered from upstream. At each time step all inventories are called. Depending on their parameters they might generate orders. After that, shipments for the existing orders are assembled and put on the readyShipments list.

Next, all the production lines are called in random order. If a production line has produced something but the downstream warehouse does not accept it, it has to wait. If its pendingOrders list is empty, a production line would wait too. If it is at the beginning of the production cycle and needs resources and pendingOrders list is not empty, it would check if there are any shipments for it ready upstream. Once all the resources are available, a production line would take them and start producing for delay number of time steps.

The order-driven model further includes a time stamped feature. In this version of the model, there is a time of delivery specified for each order. ReadyOrders is time sorted now. When an inventory receives an order, it checks if it can be satisfied from the pool of available SKUs. If it can not, it would try to reallocate resources from readyOrders in order to satisfy more urgent order at the

expense of less urgent ones. Whatever is taken from the less urgent orders is ordered from upsteam. As an order propagates upstream its time is adjusted to take delays of production lines into account. A production line would only take a shipment if the time is right. This model would be helpful when experimenting with automated markets: the price per SKU would depend among other things on how much in advance the order is placed.

Final consumers will increasingly order final SKUs by electronic shopping. This offers a multitude of important changes and opportunities for supply chain management First, if by proper pricing, consumers are induced to order some SKUs well ahead of time, this constitutes an observation of, rather than a mere prediction of, consumer demand and its fluctuations. By communicating those demand fluctuations upstream in the supply chain, better performance with lower total inventory can be achieved. Furthermore, trial marketing of innovative products, particularly with late stage differentiation, is aided as explained in international application number PCT/US99/15236, filed July 6, 1999, titled "A Method For Performing Market Segmentation and for Predicting Consumer Demand" by Eric Bonabeau, Stuart Kauffman and Richard Palmer, the contents of which are herein incorporated by reference. The use of electronic data will allow manufacture and retailers to sample efficiently for preferred sku characteristics, late stage differentiation will allow local, close to customer, product innovation. Clustering algorithms, included in above referenced patent, will allow optimal clustering of consumers around desired sets of product features, and facile production of such late stage differentiated products.

Late-stage Order Reallocation and Late Stage Product Differentiation

The present invention further includes late-stage order reallocation and late stage product differentiation. In general, the supply chain is faced with fluctuations in demand for the final SKUs at a given shelf at a final retailer, or outlet to the final consumer. The fluctuations in demand can be "smoothed" or averaged" in three different ways:

i. Over time. Thus, fluctuations per day are smoothed out over weeks or months.

ii. Over a set of final outlet (e.g. retail) stores. Thus, a given SKU may show fluctuations at a particular store, but over 2, 5,10, or 100 stores, the fluctuations in demand will tend to average out.

iii. By SKU hierarchy. Thus, a given hair care product may be bottled in seven different sizes of bottles, each a different SKU. Demand for a given SKU may fluctuate sharply, but total demand for the given hair care product, summed over all bottle sizes, may fluctuate far less.

In general, fluctuations in consumer demand make operation of the supply chain with respect to optimizing the four figures of merit difficult. Thus, the different modes of averaging over these fluctuations are of use in improving supply system performance. Notably, averaging over two or more stores, or other points in the supply chain, is a preferred means of operating the supply chain. This averaging can be achieved in a multiplicity of ways, among those, the use of convergent flow, as well as reallocation in

distributions from distribution centers to final retailers, or use of redistribution among the final retail stores themselves.

A second means of modifying supply chain functions, late product differentiation, achieves averaging over SKU hierarchies. Thus, if fluctuations for SKUs of the same hair care material in different bottle sizes occurs, late stage differentiation would place the hair care material at a distribution/manufacturing center close to one or more final retail outlets, and carry out final bottling into final SKUs at that point, in rapid adaptive response to the fluctuations in SKU demand at the set of retail outlets, or final consumers, served by that late stage differentiation point. Late stage differentiation, in our results, yields a decrease in short shipments as well as a reduction in total inventory in the system. Not yet factored into our analysis is a further aid to this reduction in inventory, namely, that if a central facility carries out only part of the total production cycle, and late stage differentiation centers carry out other parts of the production cycle, then the total time of production at any center is reduced. In turn, this leads to shorter order cycles, hence reduced step up levels in downstream inventory levels, hence smooths out and reduces total inventory. An additional virtue of late stage distribution and differentiation is that persistent product innovations can be tried in each local region of a few stores or tens of stores, to test which modifications sell well. Those that do can be tried on larger regional scales. By contrast, if all manufacture to the final SKU is carried out at a single central facility, flexible introduction of new products is relatively difficult, and a major effort.

The late stage order reallocation model of the present invention includes an agent-based simulation. Preferably, the late stage order reallocation model models a flow of a single SKU. Preferably, the model uses aggregate weekly sales data and lead times from shipping data. In the alternative, the model uses simulated demand (preferably, Poisson) that is similar in demand to actual POS data without the day-of-week variation or promotions. In another alternative, the model uses daily data. Optionally, for more accurate estimates of the effect of reallocation, the model uses information about orders that has the following characteristics:

- Daily figures
- Actual quantity desired by the DC (or store)
- Actual quantity desired by the customer of the DC (or store)
- A variety of SKU's
- As long a time period as possible (minimum four months)
- Indication of exceptional events that affected demand (promotions, holidays, etc.)

Optionally, the model may include one or more of the following assumptions:

DC's order periodically (daily, twice weekly, weekly, or once every two weeks)

Time from placing order to delivery at DC is 11 steps (5.5 days) (actual lead time varies by state from 5 to 15 days)

5 DC's all order at same time, or order and receive in a staggered manner (e.g., DC1 orders on Weds & receives on Mon, DC2 orders on Thurs and receives on Tues, etc.)

Order size determined by analysis of recent demand and a preset "acceptable out-of-stock" probability

Only constraint on orders is that order quantities must be in multiples of 5 (this number was chosen arbitrarily)

No maximum order quantity

Reallocations can be in any integer quantity (they don't have to be done in multiples of 5. E.g., if one warehouse had ordered 10 and another 15, then one could be allocated 12 and the other 13).

DC's all receive their shipment at the same time

Doing reallocation does not add any time to the delivery

Improvement relies on assumption that it is possible to reallocate orders just before final delivery among nearby DCs based on their needs.

The model further includes decision rules like the examples shown below:

Without reallocation: DC's receive quantity ordered

With reallocation: at last step in delivery, orders are reallocated among DC's minimize the probability of any DC experiencing an out-of-stock before the next delivery.

To improve its structure, the model optionally includes the following information:

- Location of particular DC's (or stores, if deliveries at stores are to be modeled)
- Frequency of ordering at those DC's (or stores)
- Trip distances (in time) between those DC's (or stores)
- Lead times for orders made by DC's (or stores)
- Estimates of extra time required at each stop to perform reallocation
- Order policies of DC's (or stores)

- Packsize for orders from DC's (or stores), per SKU, and indications of what would be reasonable packsizes for reallocation

The reallocation algorithm maximizes the expected number of SKUs sold by the time the next delivery in the pipeline arrives, or if none is in the pipeline, by the time the next delivery could arrive if it were ordered as soon as possible. Suppose that all the DCs together are to receive a total of 8 SKUs on a shipment. Then, for each DC, the probability that it sells 1 through 8 SKUs is calculated. For a particular DC, these probabilities will be non-increasing. It is a simple matter to allocate the SKUs to DCs in a manner that maximizes the expected number of SKUs sold (in addition to those the DC has in stock). This is illustrated in the table below - 8 incoming SKUs are allocated according to the shaded regions.

Pr(sell >=X)	X=1	X=2	X=3	X=4	X=5	X=6	X=7	X=8
DC1	0.9	0.85	0.8	0.75	0.69	0.57	0.5	0.4
DC2	0.95	0.89	0.79	0.68	0.58	0.3	0.1	0.05
DC3	0.85	0.74	0.73	0.7	0.59	0.2	0.05	0.01

The algorithm finds the x_i (the number of SKUs received by DC i) that maximizes $\sum_i \sum_{j=1..x_i} \Pr(\text{sell}_i \ge j)$, under the constraint that the sum of the x_i is the number of SKUs on the truck, and $x_i \ge 0$, and where sell_i is a random variable being the number of SKUs sold by DC i beyond those it has in stock.

An alternative objective is minimizing the expected number of DCs that are out-of-stock by the time the next order will or could arrive. This objective would be accomplished by finding the x_i that minimize $\sum_i \Pr(\text{sell}_i \geq x_i + 1)$, under the same constraints. In the example above, the allocations that minimize the expected number of DC's that are out-of-stock are 4,1,3. However, for most smoothly-varying demand functions (most will be much smoother than the example), the allocations made under this objective will be very similar to those made under the preceding objective.

Results determined from executing the reallocation algorithm indicate the following:

Reallocation and staggered orders work well together

- Staggered orders without reallocation has virtually no effect on out-of-stocks or on inventory at DC's (as expected)
- Reallocation without staggered orders sometimes has a moderate benefit, and sometimes hurts (when ordering frequency is once every two weeks)
- Late-stage reallocation with staggered orders can cut out-of-stocks at DC's by a factor of ten or more
- OR can reduce excess inventory in DC's by 20 to 35%, while achieving same out-of-stocks at stores.

FIGs 32-39 show results of the reallocation algorithm. FIG. 32 shows the out-of-stocks for a pool of 5 stores with standard allocation and with reallocation. FIG. 33 shows out-of-stocks for a pool of 5 stores with standard allocation and with reallocation. FIG. 34 shows excess inventory for a pool of 5 stores with standard allocation and with reallocation. FIG. 35 shows the time-in-system for a pool of 5 stores with standard allocation and with

reallocation. FIG. 36 shows out-of-stocks for standard allocation and reallocation in a log-log plot. FIG. 37 shows out-of-stocks for standard allocation and for reallocation for 3, 5, and 10 stores. FIG. 38 shows excess inventory for standard allocation and for reallocation for 3, 5, and 10 stores. FIG. 39 shows the time-in-system for standard allocation and reallocation for 3, 5, and 10 stores.

Preferably, the profits of the late stage order reallocation model is affected by one or more of the following: profit on net of purchases/sales, order setup costs, holding costs and short penalty. Preferably, stores order based on preset order size and re-order point (from back-of-envelope calculations for economic order quantify and re-order point). Preferably, each process has a batch size and units must flow through each process in multiples of its batch size.

The late stage order reallocation model further includes the ability to construct a landscape representation from a parameter space to a results space. The parameter space include a trigger level (re-order point) and/or a batch size (units can only flow through processes in multiples of batch size.) FIG. 40 shows a landscape of trigger level to profit for the reallocation model. FIGs. 41 and 42 show landscapes of batch size to total profit. These results show that the trigger level landscape is smooth for a simple supply chain but a batch size landscape is rugged.

The reallocation algorithms include a tau-search for a good operating configuration. The tau-search is a landscape search that is controlled by tau (i.e. the number of sites trying new settings simultaneously). Exemplary tau search parameters include the following: start with all batch sizes equal to one; generate a new configuration to

try, by moving 1 to 5 (tau) batch sizes by up to Maxjump units; evaluate the profitability of the configuration; if it is better, accept the new configuration and search from there; and if it is worse, reject the old configuration and try the next step from the old one.

FIGs. 43-48 show results for the tau-search algorithm. FIG. 43 shows an example progress of a tau-search when tau is one and maxjump is six. FIG. 44 shows an example progress of a tau-search when tau is five and maxjump is six. FIG. 45 shows an example progress for a tau-search when tau is one and maxjump is one. FIG. 46 shows the fitness of the best configuration that was found. FIG. 47 shows a chart of the steps that were accepted in the search. FIG. 48 shows the tries to reach the best configuration that was found.

FIG. 49 discloses a representative computing system 4910 in conjunction with which the embodiments of the present invention may be implemented and executed. Computing system 4910 may be a personal computer, workstation, or a larger system such as a minicomputer. However, one skilled in the art of computer systems will understand that the present invention is not limited to a particular class or model of computer.

As shown in FIG. 49,representative computing system 4910 includes a central processing unit (CPU) 4912, a memory unit 4914, one or more storage devices 4916, an input device 4918, an output device 4920, and communication interface 4922. A system bus 4924 is provided for communications between these elements. Computing system 4910 may additionally function through use of an operating system such as Windows, DOS, or UNIX. However, one skilled in the art of computing systems will understand that the present invention

is not limited to a particular configuration or operating system.

Storage devices 4916 may illustratively include one or more floppy or hard disk drives, CD-ROMs, DVDs, or tapes. Input device 4918 comprises a keyboard, mouse, microphone, or other similar device. Output device 4920 is a computer monitor or any other known computer output device. Communication interface 4922 may be a modem, a network interface, or other connection to external electronic devices, such as a serial or parallel port

While the above invention has been described with reference to certain preferred embodiments, the scope of the present invention is not limited to these embodiments. One skill in the art may find variations of these preferred embodiments which, nevertheless, fall within the spirit of the present invention, whose scope is defined by the claims set forth below.

minimize the expected number of sites that are out of inventory.

67. A method for managing a supply chain as in claim 28 wherein said performance space comprises one or more members of the group comprising total inventory in the supply chain, total time of inventory in the supply chain, short shipments and cost.

68. Computer executable software code stored on a computer readable medium, the code for managing a supply chain, the code comprising:

code to define one or more models for the one or more sites having one or more parameters, said one or more parameters having one or more settings;

code to define a performance space; and

code to search for one or more optimal values of said settings of said one or more parameters comprising:

code to change at least one of said settings at one or more of the sites; and

code to simulate said one or more models to generate one or more corresponding values in said performance space.

69. A programmed component for managing a supply chain comprising at least one memory having at least one region storing computer executable program code and at least one processor for executing the program code stored in said memory, wherein the program code comprises:

code to define one or more models for the one or more sites having one or more parameters, said one or more parameters having one or more settings;

Claims

1. A method for managing at least one supply chain comprising the steps of:

defining at least one model having a parameter space to represent the supply chain;

defining a performance space;

defining at least one policy for the supply chain; and

softening said at least one policy.

2. A method for managing at least one supply chain as in claim 1 wherein said parameter space comprises a shipping capacity.

3. A method for managing at least one supply chain as in claim 1 wherein said parameter space comprises an inventory capacity.

4. A method for managing at least one supply chain as in claim 1 wherein said parameter space comprises a batch size.

5. A method for managing at least one supply chain as in claim 1 wherein said parameter space comprises a production delay.

6. A method for managing at least one supply chain as in claim 1 wherein said at least one model is a graph.

7. A method for managing at least one supply chain as in claim 6 wherein said graph is a bipartite graph.

8. A method for managing at least one supply chain as in claim 6 wherein said graph comprises one or more nodes to represent one or more storage facilities.

9. A method for managing at least one supply chain as in claim 6 wherein said graph comprises one or more nodes to represent one or more production facilities.

10. A method for managing at least one supply chain as in claim 6 wherein said graph comprises one or more nodes to represent one or more transportation facilities.

11. A method for managing at least one supply chain as in claim 1 wherein said at least one policy comprises one or more conditions for commencing a production line.

12. A method for managing at least one supply chain as in claim 1 wherein said softening said at least one policy step comprises the step of commencing a production line when a subset M-K of required resources are available.

13. A method for managing at least one supply chain as in claim 12 wherein the probability of commencing a production line is:

$$p=\exp(-K^2/2\ \text{sigma}^2)$$

wherein

sigma is a constant.

14. A method for managing at least one supply chain as in claim 1 wherein said at least one policy comprises one or more conditions for commencing transportation.

15. A method for managing at least one supply chain as in claim 1 wherein said softening said at least one policy step comprises the step of commencing transportation upon receipt of a subset M-K of transportation capacity.

16. A method for managing at least one supply chain as in claim 15 wherein the probability of commencing a transportation is:

$$p=\exp(-K^2/2\ \mathrm{sigma}^2)$$

wherein

sigma is a constant.

17. A method for managing at least one supply chain as in claim 1 wherein said softening step widens an optimal range in said parameter space.

18. A method for managing at least one supply chain as in claim 1 wherein said parameter space comprises at least one noisy parameter.

19. A method for managing at least one supply chain as in claim 18 wherein said softening step makes one or more optimal points more stable with respect to the noise.

20. A method for managing at least one supply chain as in claim 1 wherein said performance space comprises a level of inventory of a final product.

21. A method for managing at least one supply chain as in claim 1 wherein said performance space comprises a level of short shipments.

22. A method for managing at least one supply chain as in claim 1 wherein said performance space comprises an amount of time needed to produce a resource.

23. A method for managing at least one supply chain as in claim 1 further comprising the step of simulating said at least one model.

24. A method for managing at least one supply chain as in claim 23 further comprising the step of repeating said simulating step to produce a landscape representation.

25. A method for managing at least one supply chain as in claim 24 wherein said softening said at least one policy step makes the landscape representation less rugged.

26. Computer executable software code stored on a computer readable medium, the code for managing at least one supply chain the code comprising:

code to define at least one model having a parameter space to represent the supply chain;

code to define a performance space;

code to define at least one policy for the supply chain; and

code to soften said at least one policy.

27. A programmed component for managing at least one supply chain comprising at least one memory having at

least one region storing computer executable program code and at least one processor for executing the program code stored in said memory, wherein the program code comprises:

code to define at least one model having a parameter space to represent the supply chain;

code to define a performance space;

code to define at least one policy for the supply chain; and

code to soften said at least one policy.

28. A method for managing a supply chain involving one or more sites comprising the steps of:

defining one or more models for the one or more sites having one or more parameters, said one or more parameters having one or more settings;

defining a performance space; and

searching for one or more optimal values of said settings of said one or more parameters comprising the steps of:

changing at least one of said settings at one or more of the sites; and

simulating said one or more models to generate one or more corresponding values in said performance space.

29. A method for managing a supply chain as in claim 28 wherein said searching step further comprises the step of repeating said changing said one or more settings step and said simulating said one or more models step.

30. A method for managing a supply chain as in claim 22 wherein said one or more parameters comprise a re-order point.

31. A method for managing a supply chain as in claim 28 wherein said one or more parameters comprise a batch size.

32. A method for managing a supply chain as in claim 28 wherein said one or more parameters comprise a degree of information sharing.

33. A method for managing a supply chain as in claim 28 wherein said information sharing comprises advance notice of promotions.

34. A method for managing a supply chain as in claim 28 wherein said one or more parameters comprise a number of stock boys.

35. A method for managing a supply chain as in claim 28 wherein said performance space comprises a profit.

36. A method for managing a supply chain as in claim 28 wherein said one or more settings are changed by a maximum amount, maxjump.

37. A method for managing a supply chain as in claim 36 further comprising the step of repeating said changing step and said simulating step for different values of said maxjump.

38. A method for managing a supply chain as in claim 36 further comprising the step of determining an optimal value for maxjump.

39. A method for managing a supply chain as in claim 28 wherein a predetermined number, tau, of said one or more settings are changed simultaneously.

40. A method for managing a supply chain as in claim 39 further comprising the step of repeating said changing step and said simulating step for different values of said tau.

41. A method for managing s supply chain as in claim 39 further comprising the step of determining an optimal value for said tau.

42. A method for managing a supply chain as in claim 28 wherein said model is agent based.

43. A method for managing a supply chain as in claim 28 wherein said model comprises a demand.

44. A method for managing a supply chain as in claim 43 wherein said demand is determined from actual data.

45. A method for managing a supply chain as in claim 43 wherein said demand is determined through simulation.

46. A method for managing a supply chain as in claim 45 wherein said simulated demand has a Poisson distribution.

47. A method for managing a supply chain as in claim 28 wherein said model comprises one or more assumptions.

48. A method for managing a supply chain as in claim 47 wherein said one or more assumptions comprise one or more members of the group consisting of profit on net purchases/sales, order setup costs, holding cost, and a penalty for being short at stores.

49. A method for managing a supply chain as in claim 28 wherein the one or more sites comprises one or more members of the group consisting of warehouses, production facilities, distribution centers, stores and transportation facilities.

50. A method for managing a supply chain as in claim 28 wherein said model comprises one or more boundary condition representations.

51. A method for managing a supply chain as in claim 50 wherein said one or more boundary condition representations comprises a supply boundary condition representation.

52. A method for managing a supply chain as in claim 51 wherein said supply boundary condition representation comprises an inventory with an infinite supply of goods and a production line.

53. A method for managing a supply chain as in claim 50 wherein said one or more boundary condition

representations comprises a demand boundary condition representation.

54. A method for managing a supply chain as in claim 51 wherein said demand boundary condition representation comprises a production line and an inventory that can accept an infinite number of goods.

55. A method for managing a supply chain as in claim 28 wherein said parameters comprise a multiplicity.

56. A method for managing a supply chain as in claim 28 wherein said parameters comprise a delay.

57. A method for managing a supply chain as in claim 28 further comprising the step of transforming said one or more models to one or more simplified models.

58. A method for managing a supply chain as in claim 57 wherein said simplified model comprises a supply boundary condition representation, a demand boundary condition representation and a generic plant representation.

59. A method for managing a supply chain as in claim 58 wherein said generic plant comprises a production line.

60. A method for managing a supply chain as in claim 57 wherein said simulating step simulates said one or more simplified models for faster performance.

61. A method for managing a supply chain as in claim 28 wherein said changing one or more settings step comprises the step of:

changing connectivity among the sites.

62. A method for managing a supply chain as in claim 61 wherein said changing connectivity step comprises the step of increasing an amount of convergent flow among the sites whereon one or more of the sites have input from two or more of said sites.

63. A method for managing a supply chain as in claim 28 wherein the sites comprise one or more members of the group consisting of distribution centers, stores, production lines and warehouses.

64. A method for managing a supply chain as in claim 28 wherein said changing one or more settings step comprises the step of distributing goods among a group of sites.

65. A method for managing a supply chain as in claim 64 wherein said distributing goods step is performed to maximize $\sum_i \sum_{j=1..x_i} \Pr(\text{sell}_i \geq j)$, wherein the sum of the x_i is the number of SKUs, $x_i \geq 0$, and where sell_i is a random variable being the number of SKUs sold by DC i beyond those it has in stock.

66. A method for managing a supply chain as in claim 64 wherein said distributing goods step is performed to

code to define a performance space; and

code to search for one or more optimal values of said settings of said one or more parameters comprising:

code to change at least one of said settings at one or more of the sites; and

code to simulate said one or more models to generate one or more corresponding values in said performance space.

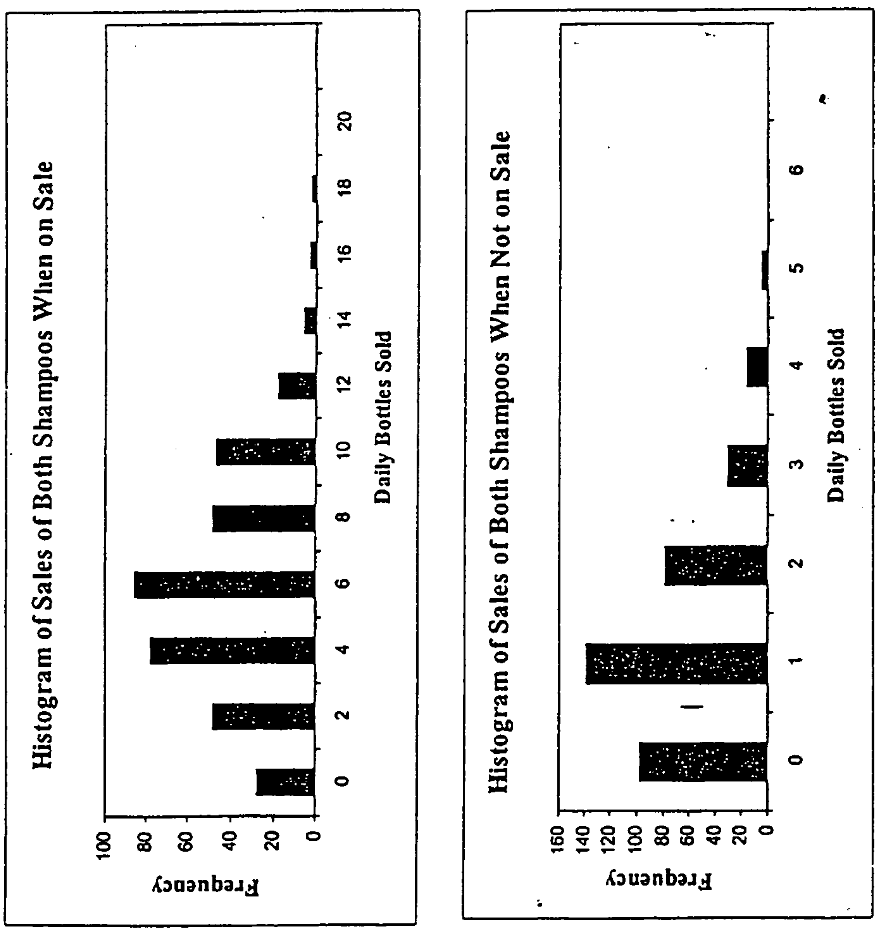

Histogram of Sales of Both Shampoos When on Sale
Frequency
100 80 60 40 20 0
0 2 4 6 8 10 12 14 16 18 20
Daily Bottles Sold
Histogram of Sales of Both Shampoos When Not on Sale
Frequency
160 140 120 100 80 60 40 20 0
0 1 2 3 4 5 6
Daily Bottles Sold

FIG. 1

Stores Choose Sales Randomly

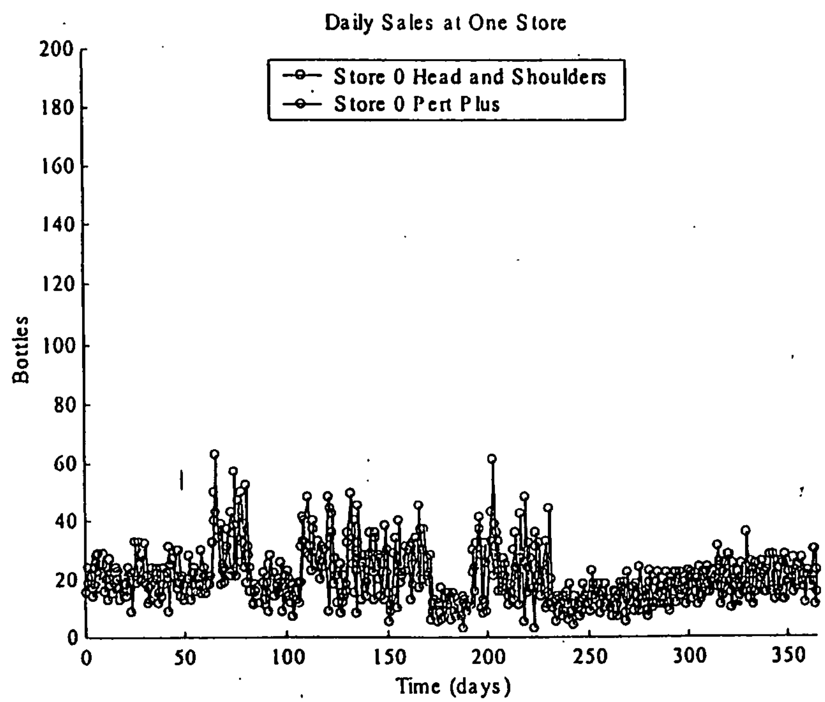


FIG. 2

Shelf Inventory - Random Sales - No Info Sharing

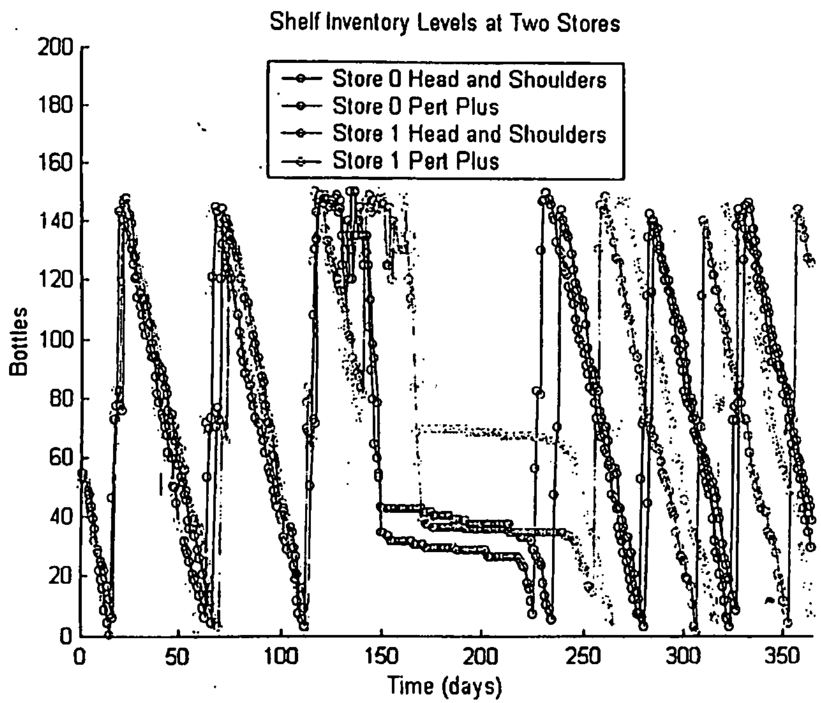


FIG. 3

DC Inventory - Random Sales - No Info Sharing

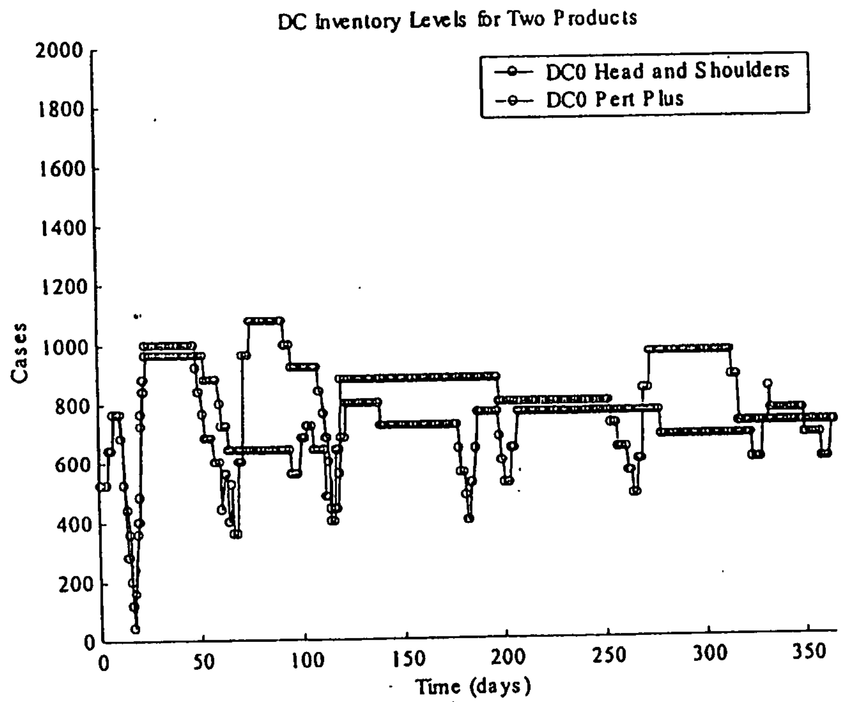


FIG. 4

Daily Sales - Random Sales

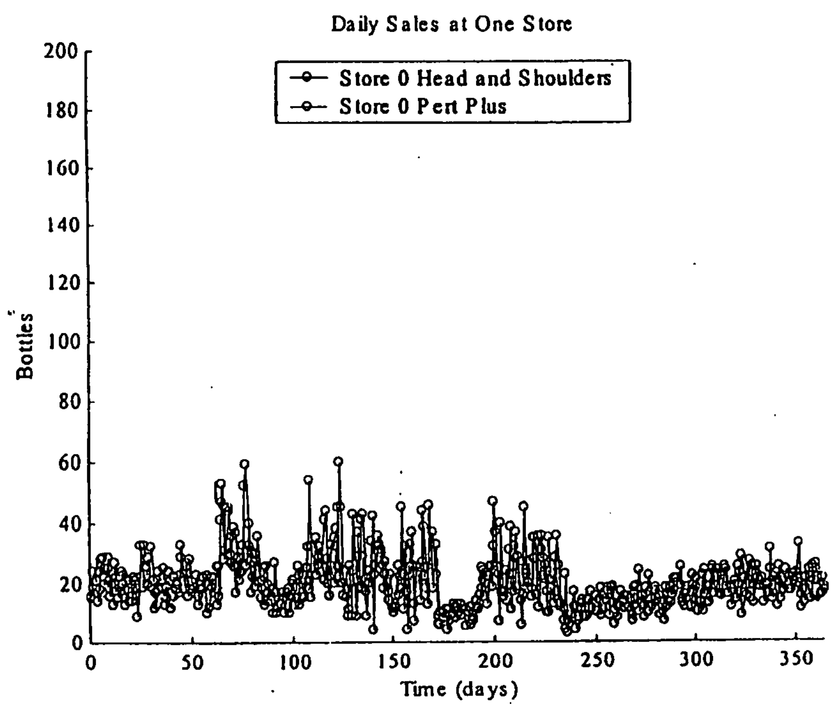


FIG. 5

Shelf Inventory - Random Sales - Info Sharing

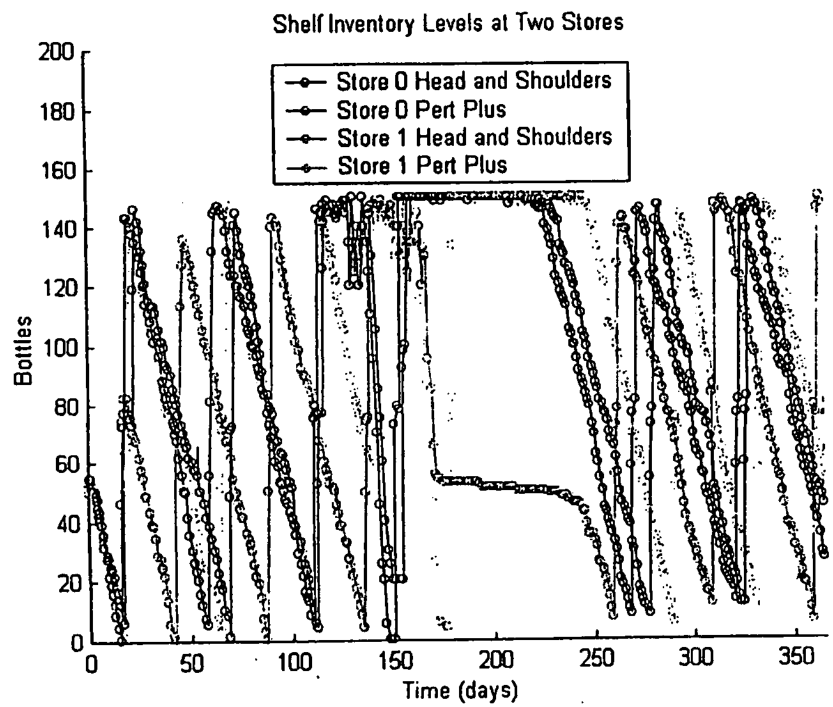


FIG. 6

DC Inventory - Random Sales - Info Sharing

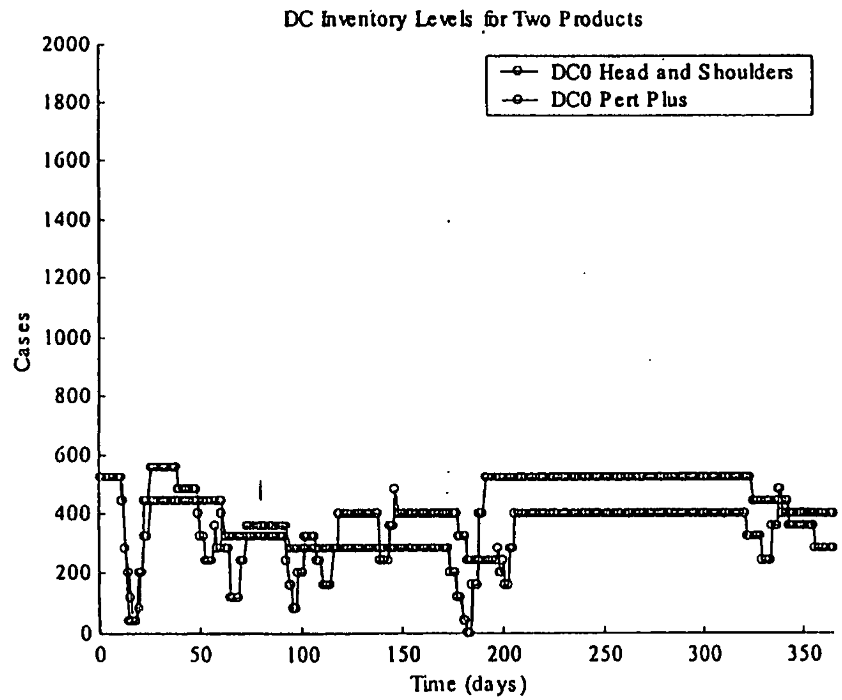


FIG. 7

Actual Point of Sale Data

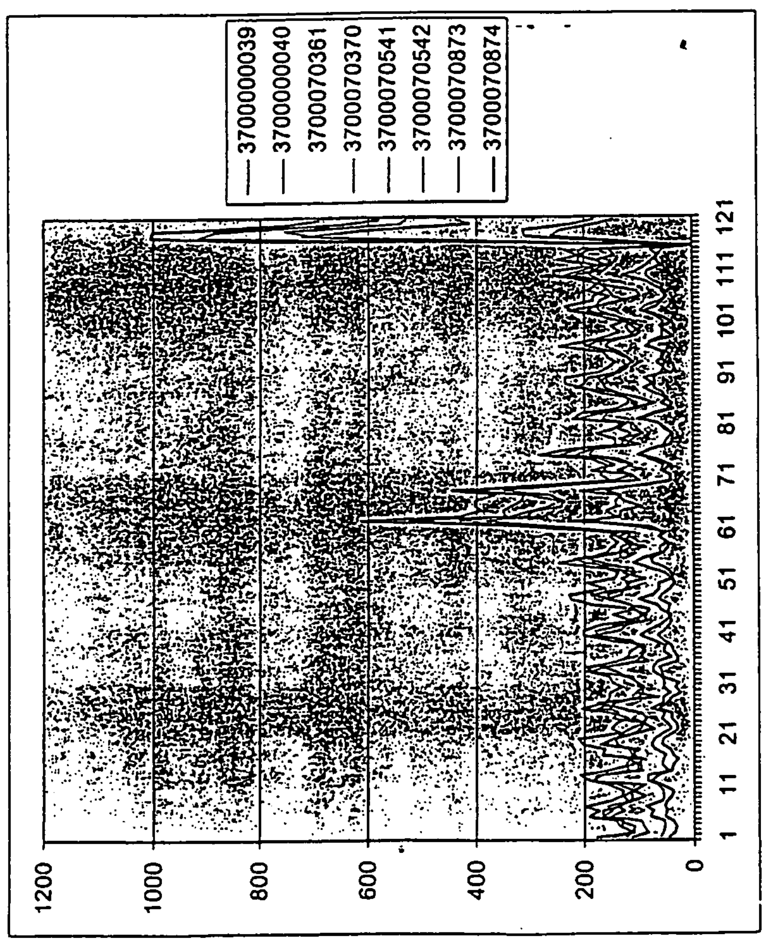


FIG. 8

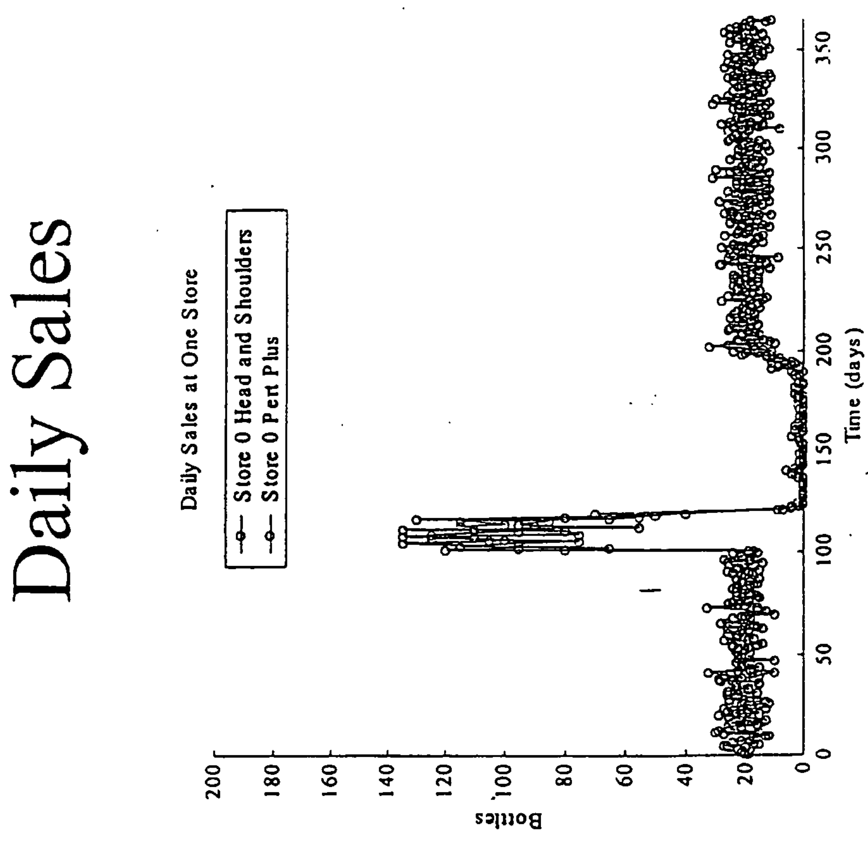

Daily Sales
Daily Sales at One Store
Store 0 Head and Shoulders
Store 0 Pert Plus
Bottles
200
180
160
140
120
100
80
60
40
20
0
0
50
100
150
200
250
300
350
Time (days)

FIG. 9

Shelf Inventory - No Info Sharing

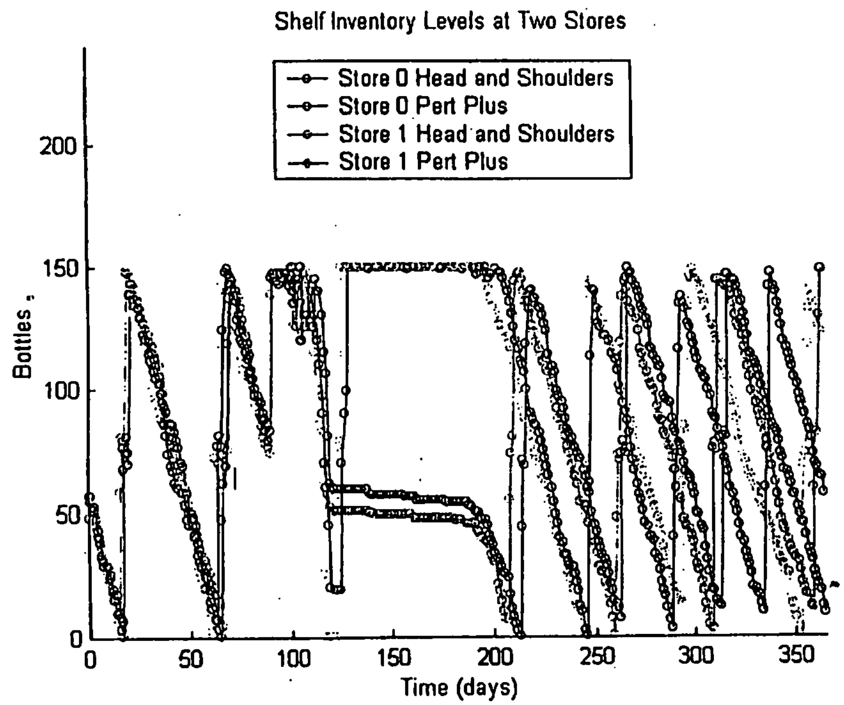


FIG. 10

Retail DC - No Info Sharing

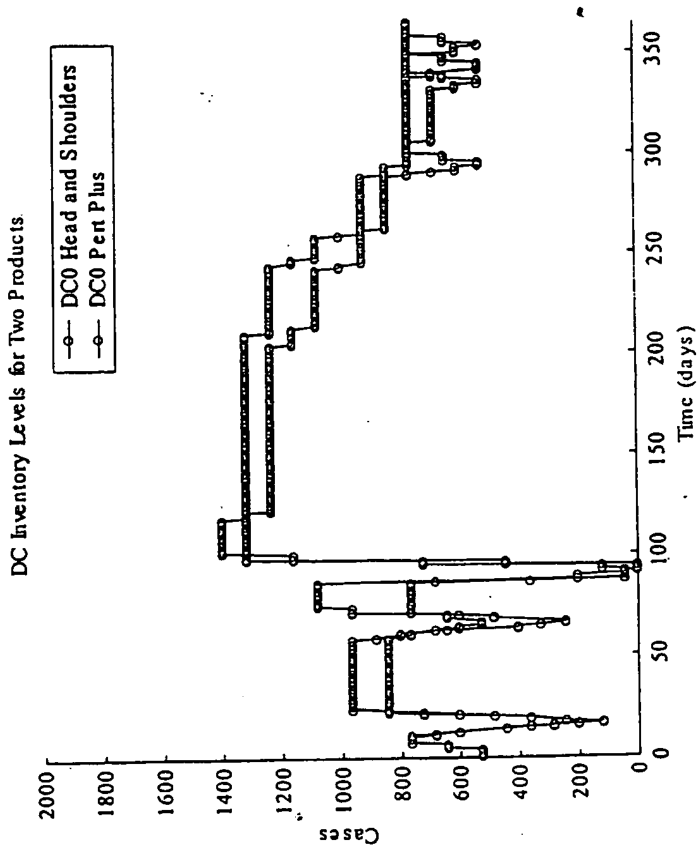


FIG. 11

Simulated Data - Daily Sales

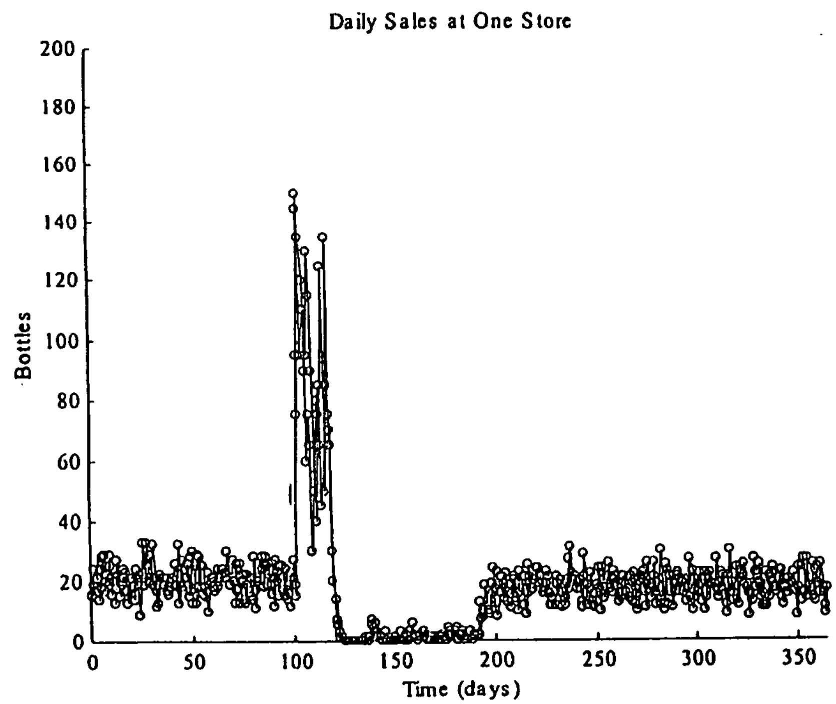


FIG. 12

Store Shelf - Info Sharing

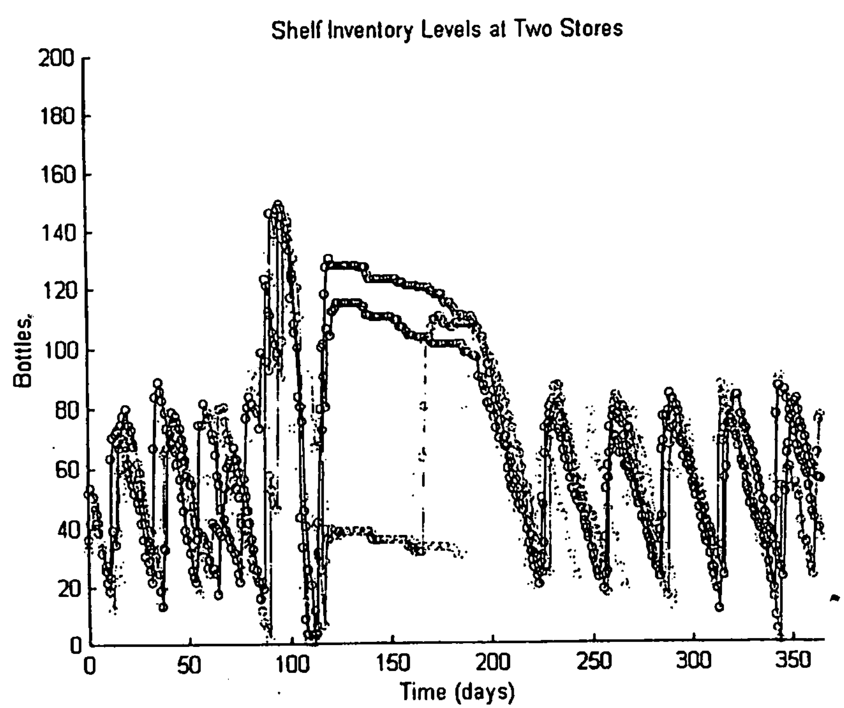


FIG. 13

Retail DC - Info Sharing

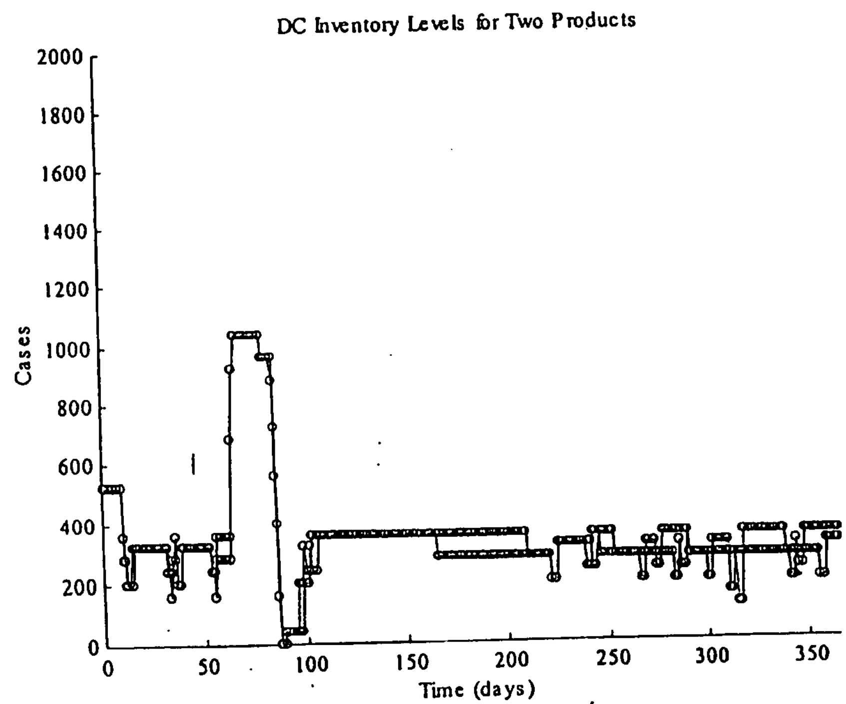


FIG. 14

	Total Sales	Out of Stock	Inventory
Info	14979.3125	2.125	414601.5
No Info	14936.75	4.895833333	786748.1
	average over big truck size		
bt = 120	13492.38	109.5188	354727.8
bt = 240	13518.16	109.7188	411841.9
bt = 360	13524.34	108.0229	470268.1
	average over little truck size		
lt = 30	13308.23	122.5167	407790.5
lt = 40	13514.36	110.075	409117.8
lt = 50	13541.66	106.7167	409759.2
lt = 60	13682.26	97.03889	422449.6

FIG. 15

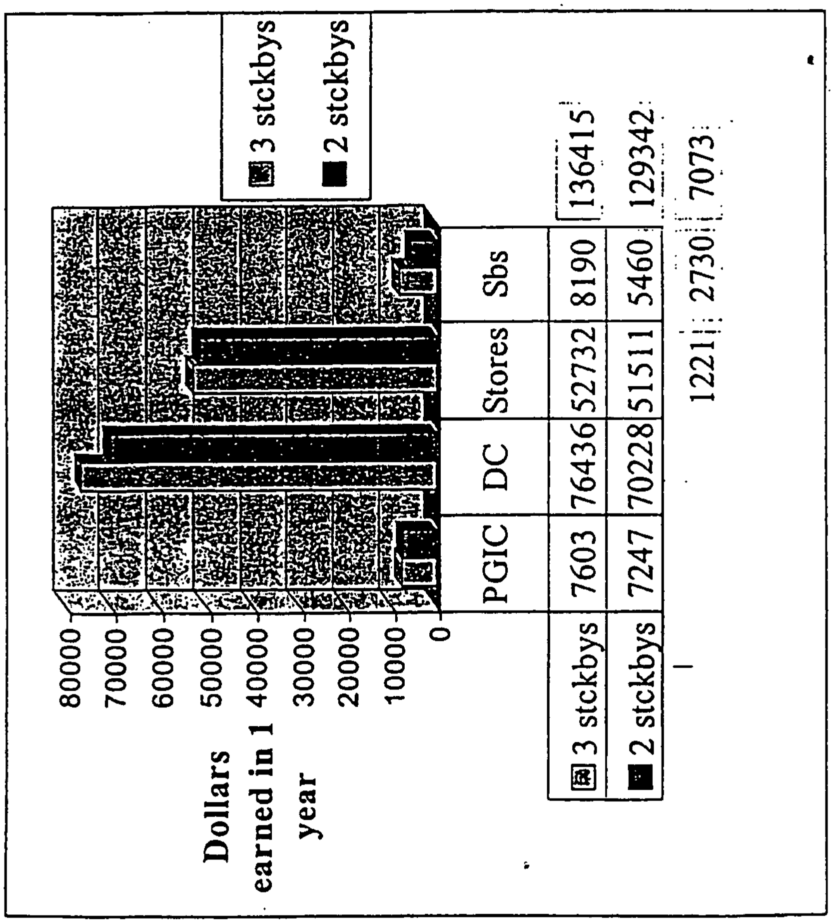

Dollars earned in 1 year
80000
70000
60000
50000
40000
30000
20000
10000
0
3 stckbys
2 stckbys
PGIC
DC
Stores
Sbs
3 stckbys 7603 76436 52732 8190 136415
2 stckbys 7247 70228 51511 5460 129342
1221 2730 7073

FIG. 16

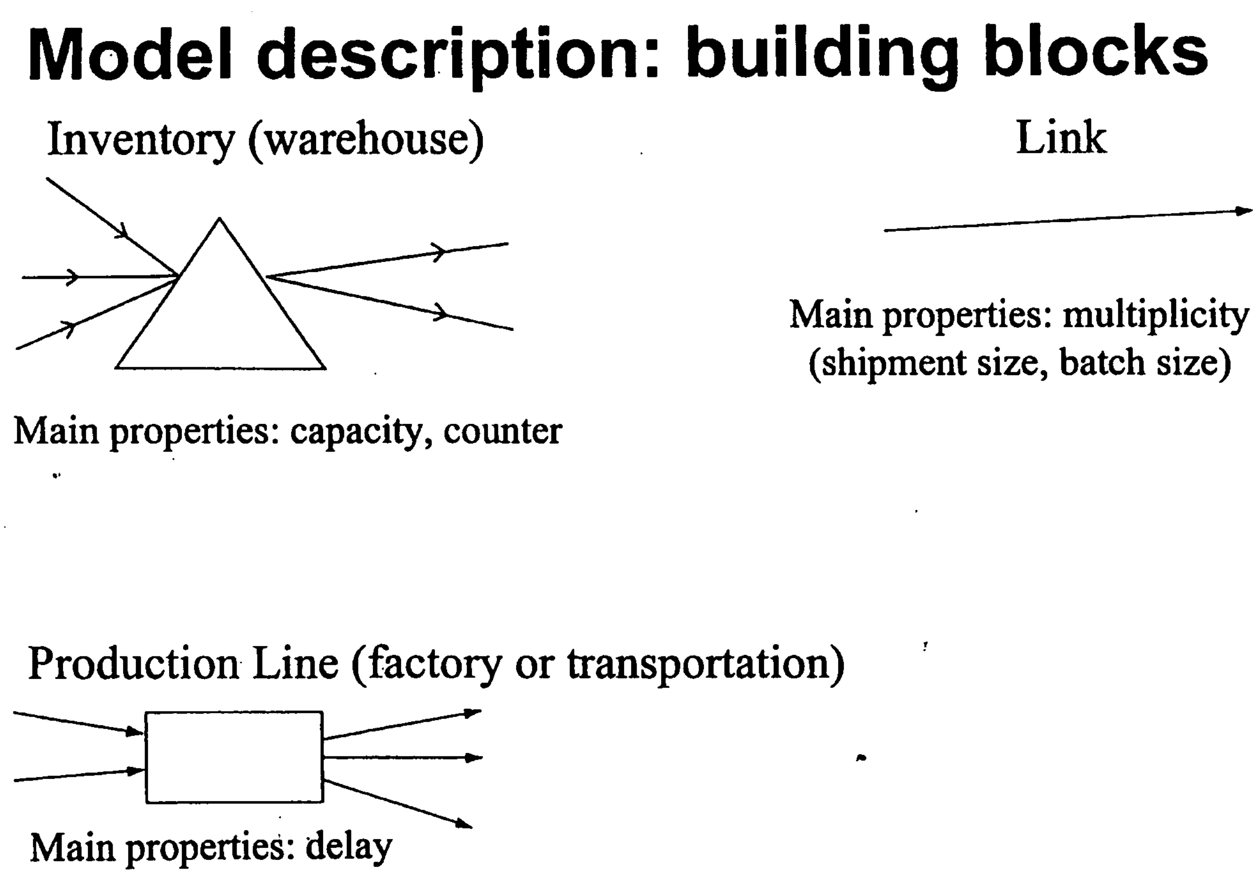

Model description: building blocks
Inventory (warehouse)
Main properties: capacity, counter
Link
Main properties: multiplicity
(shipment size, batch size)
Production Line (factory or transportation)
Main properties: delay

FIG. 17

Model description: simulating boundary conditions

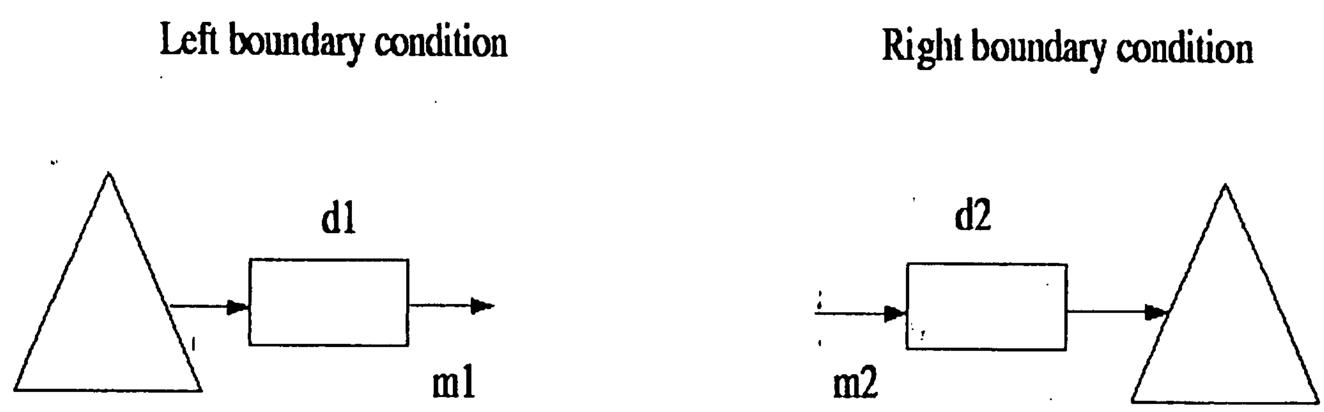
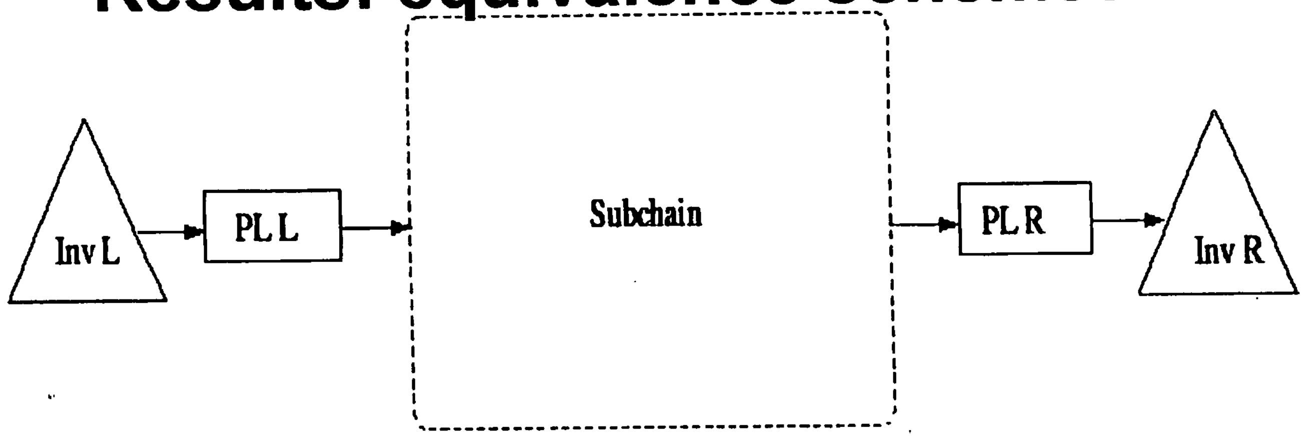


FIG. 18

Results: equivalence schemes



$$m_L = m_R = m$$

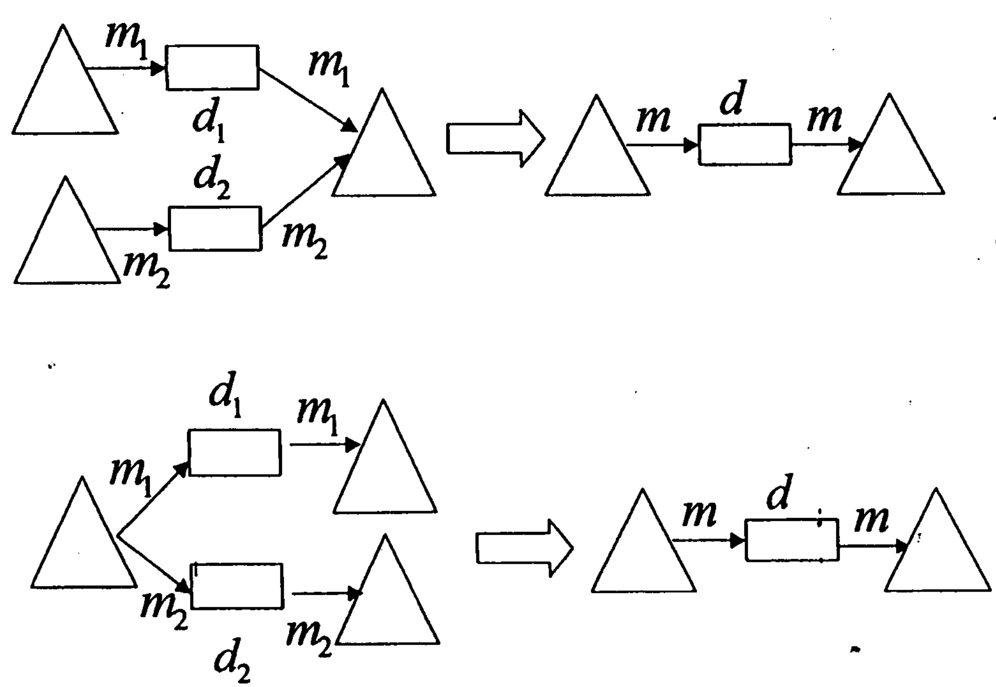
$$d_L = d_R = d$$

$$J_{tot} = \frac{m}{d}$$

This resembles very much an electrical circuit with the multiplicity being an electric charge. Is there an analogue of Kirchoff's rules?

FIG. 19

Results: equivalence schemes



$$\frac{m}{d} = \frac{m_1}{d_1} + \frac{m_2}{d_2}$$

$$m = \frac{j_1}{j} m_1 + \frac{j_2}{j} m_2$$

$$j = j_1 + j_2$$

$$j_i = \frac{m_i}{d_i}$$

FIG. 20

Results: equivalence schemes

Using these equivalence relations, we can, in many cases, reduce a supply chain of complicated topology to a simple linear generic supply chain which consists of left and right boundary constructions and a generic plant in the middle:

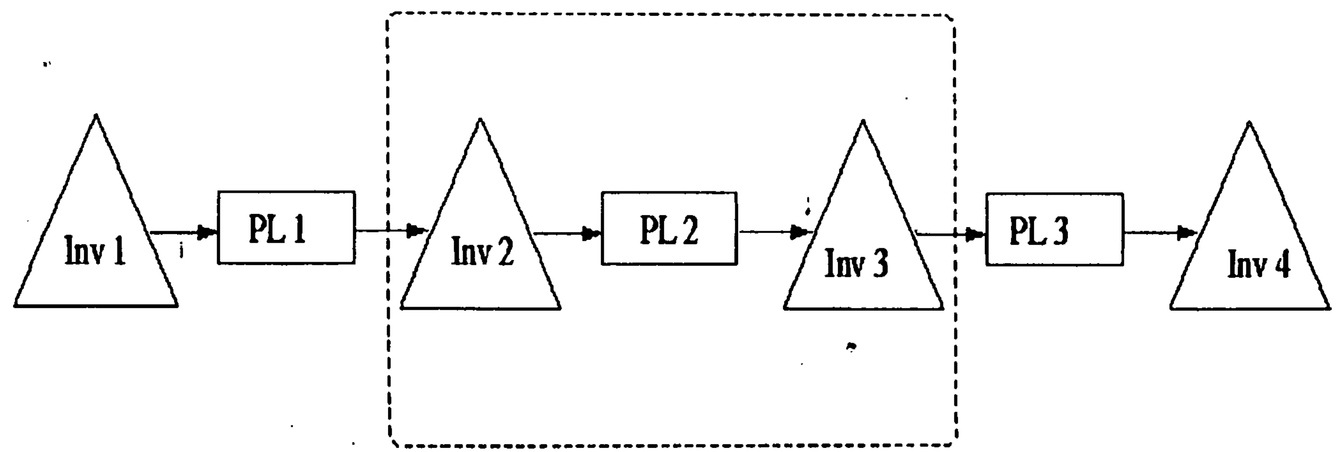


FIG. 21

Results: equivalence schemes at work

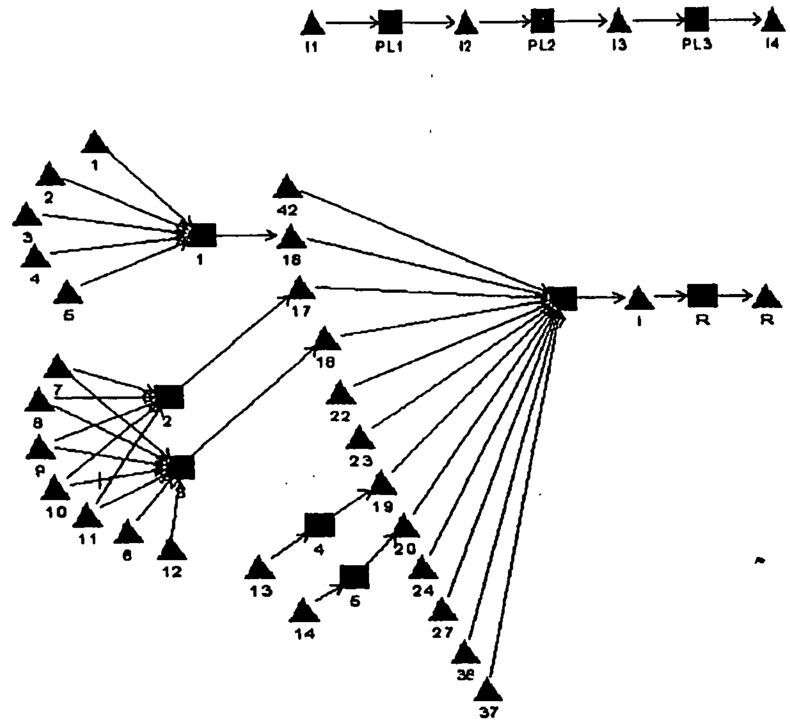


FIG. 22

Results: equivalence schemes at work

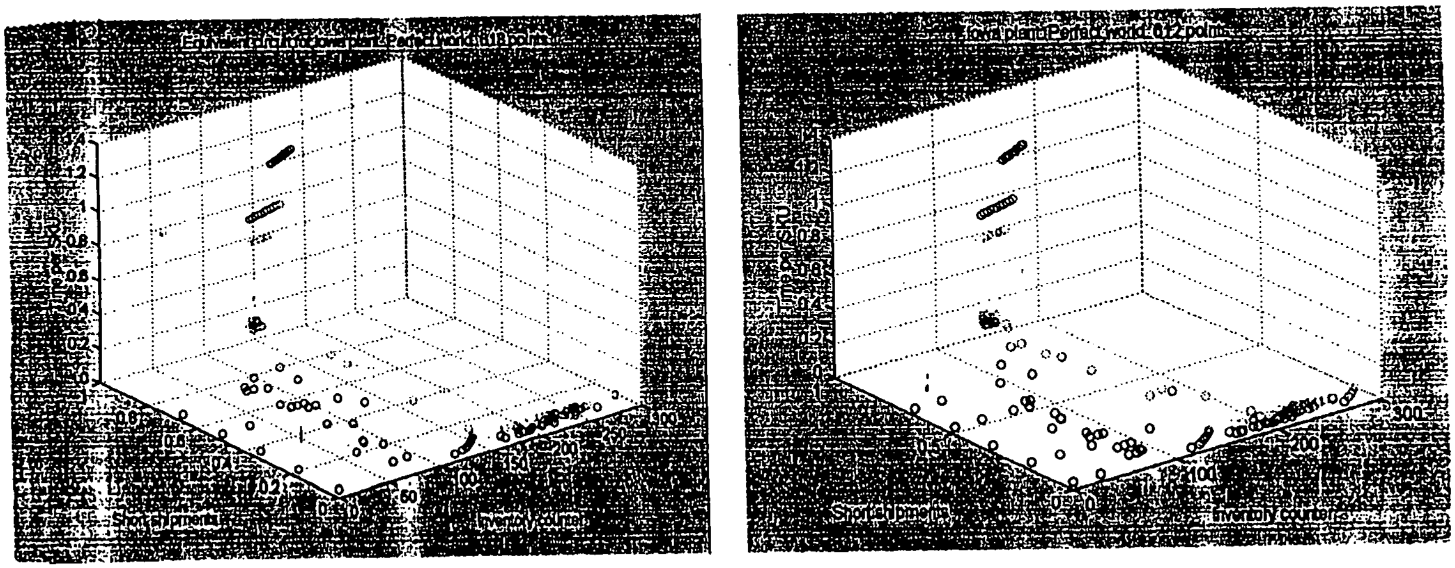
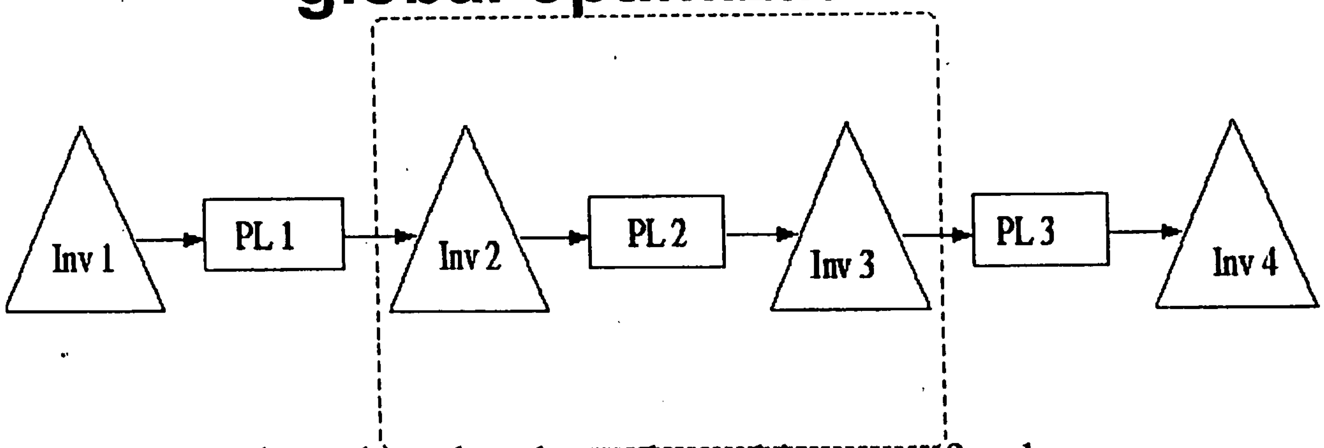


FIG. 23

Results: bottlenecks, local versus global optimization



$d_2 > d_3$ bottleneck at the plant, short shipments for the consumer

$d_2 < d_3$ bottleneck at the consumer, growing inventory at the plant

$d_2 = d_3$ global optimum

FIG. 24

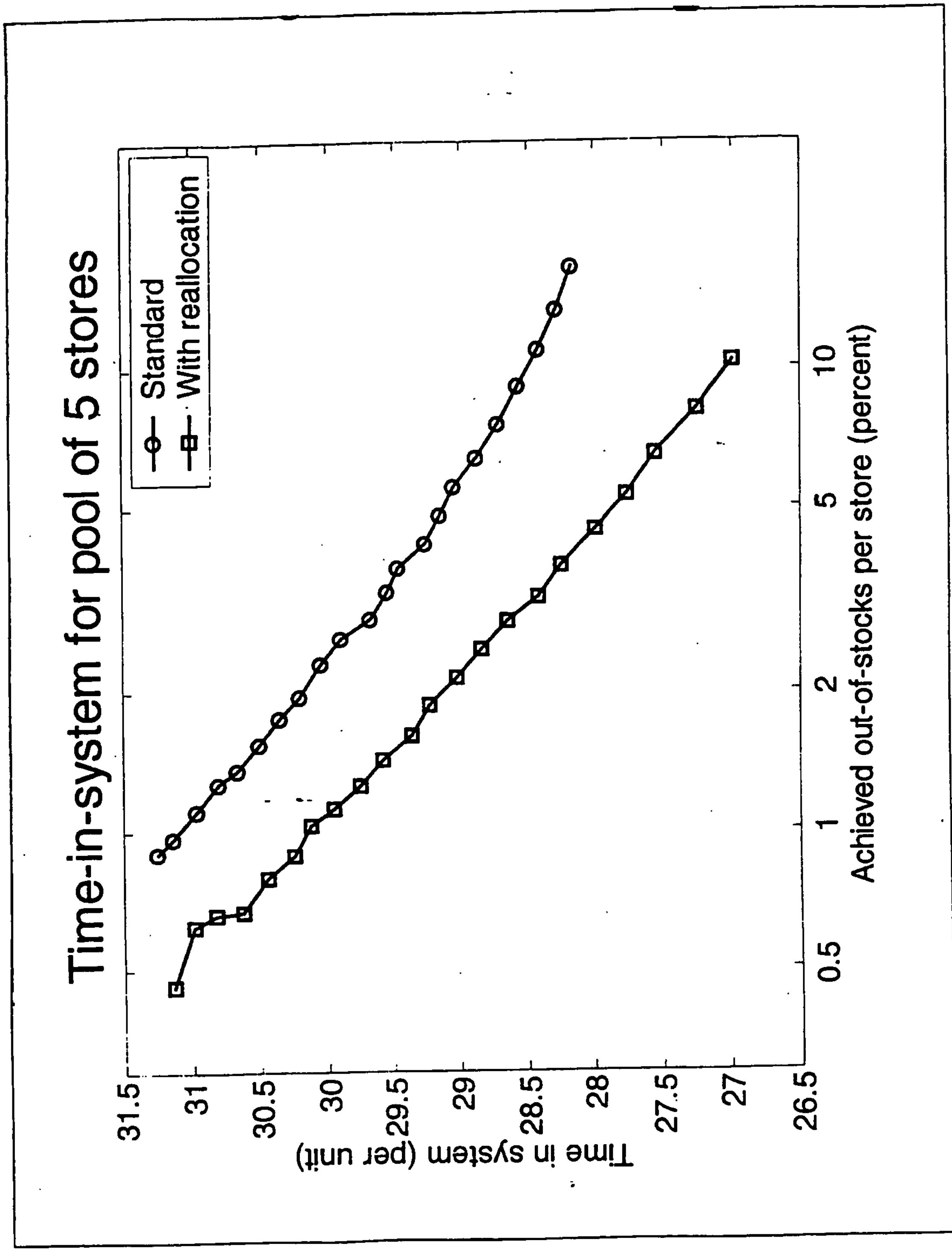

Time-in-system for pool of 5 stores
Standard
With reallocation
Time in system (per unit)
31.5
31
30.5
30
29.5
29
28.5
28
27.5
27
26.5
0.5
1
2
5
10
Achieved out-of-stocks per store (percent)

FIG. 35

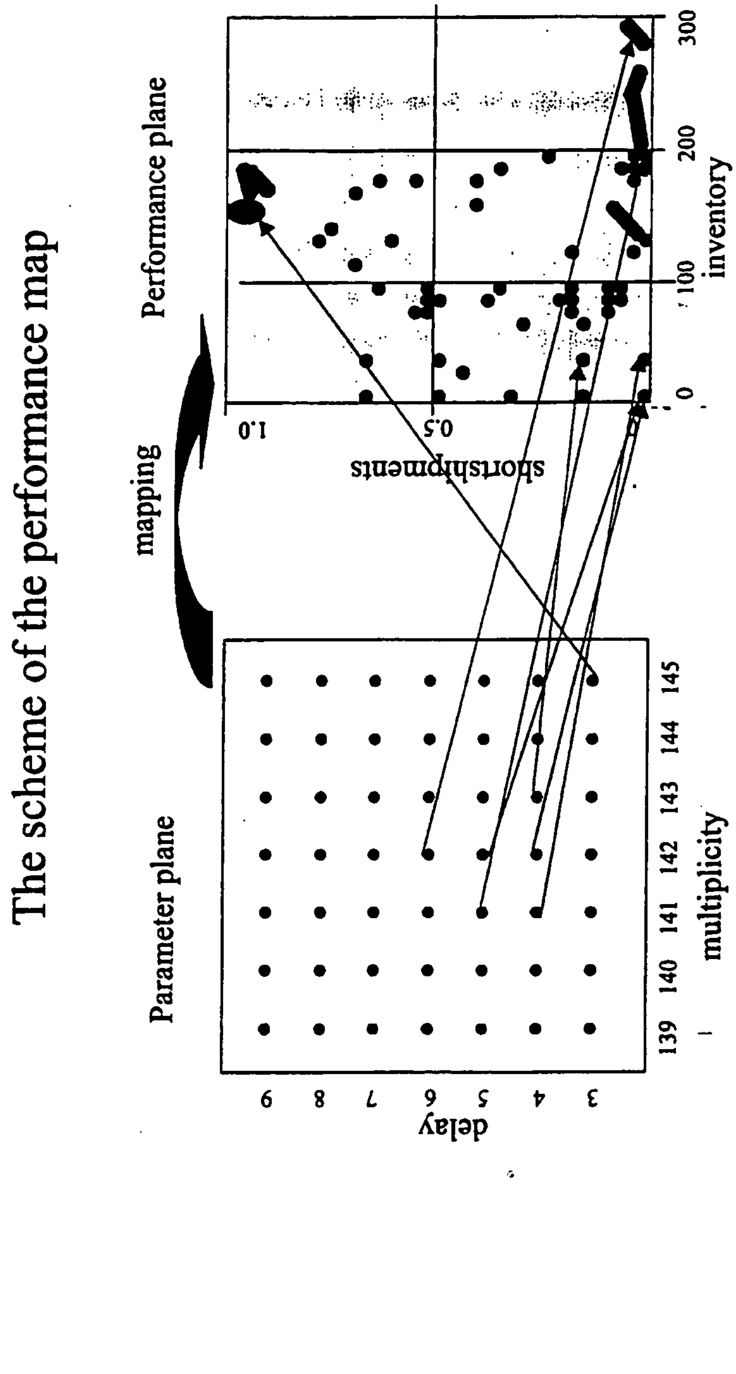

The scheme of the performance map
Parameter plane
mapping
Performance plane
delay
3
4
5
6
7
8
9
139
140
141
142
143
144
145
multiplicity
shortshipments
1.0
0.5
0
100
200
300
inventory

FIG. 25

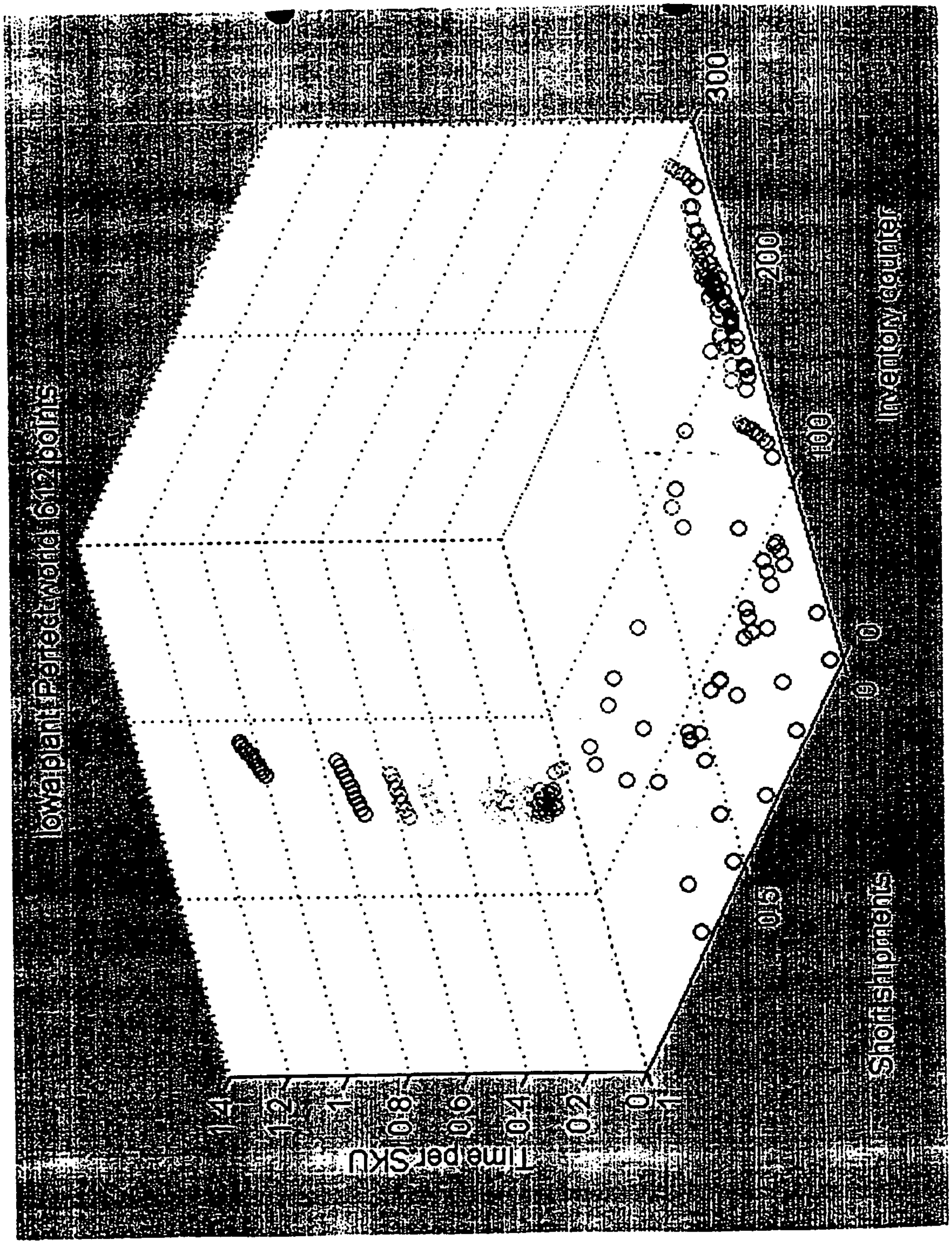

Iowa plant: Perfect world: 612 points
Time per SKU
1.4
1.2
1
0.8
0.6
0.4
0.2
0
Inventory counter
300
200
100
0
Short shipments
0
0.5
1

FIG 26

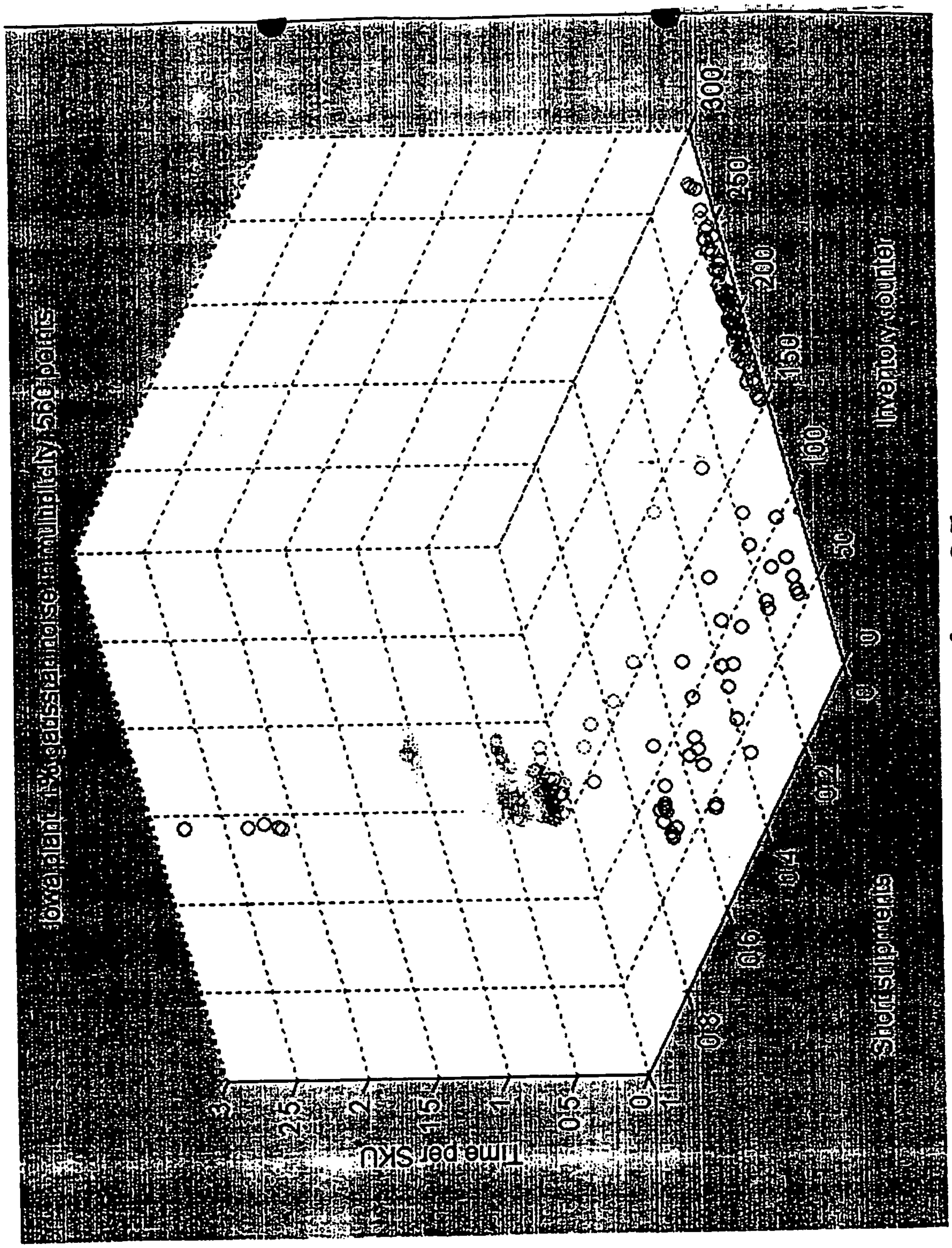

Time per SKU
Inventory counter
Short shipments
0
0.5
1
1.5
2
2.5
3
50
100
150
200
250
300
0.2
0.4
0.6
0.8
1

FIG 27

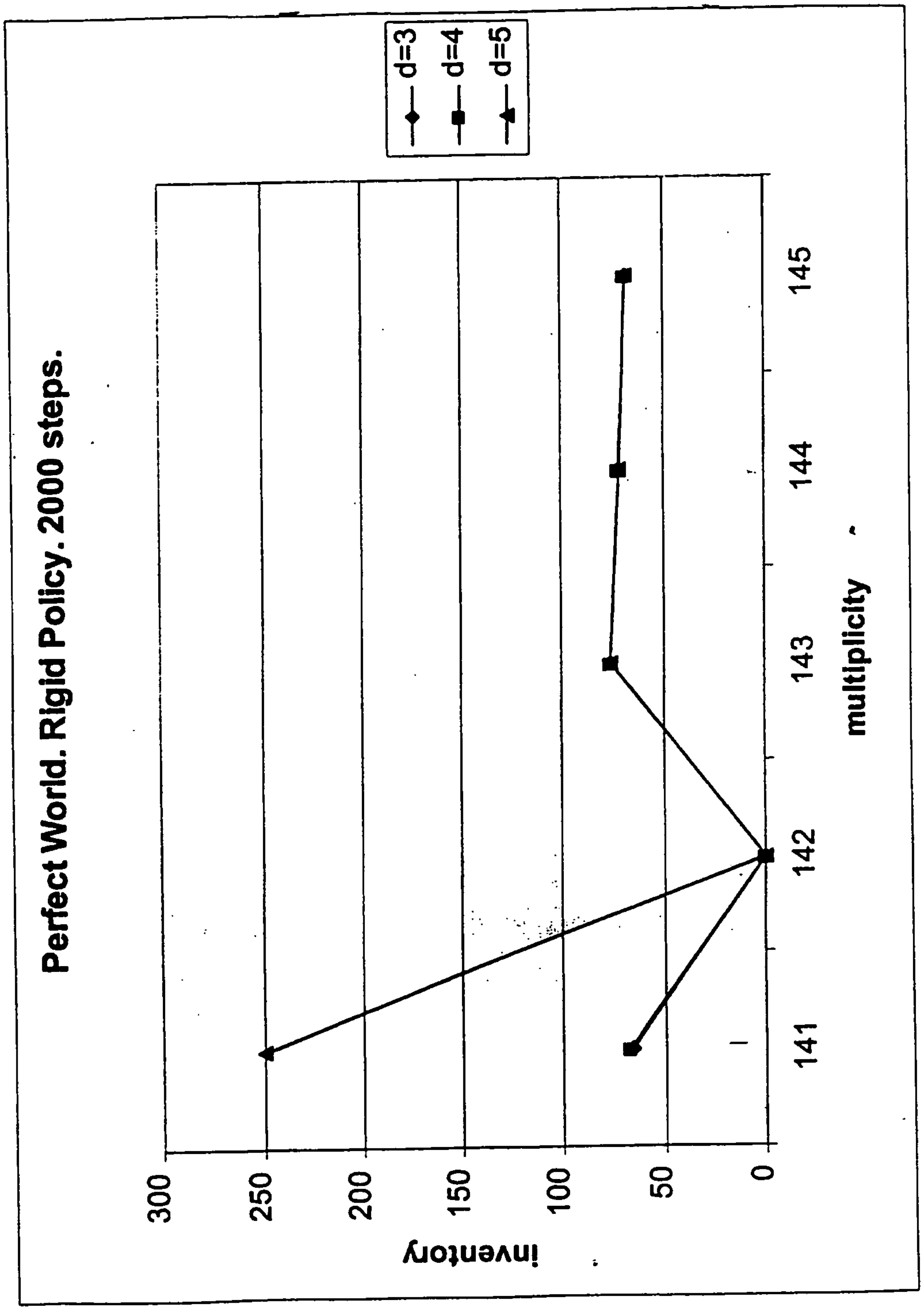

Perfect World. Rigid Policy. 2000 steps.
d=3
d=4
d=5
inventory
300
250
200
150
100
50
0
141
142
143
144
145
multiplicity

FIG. 28

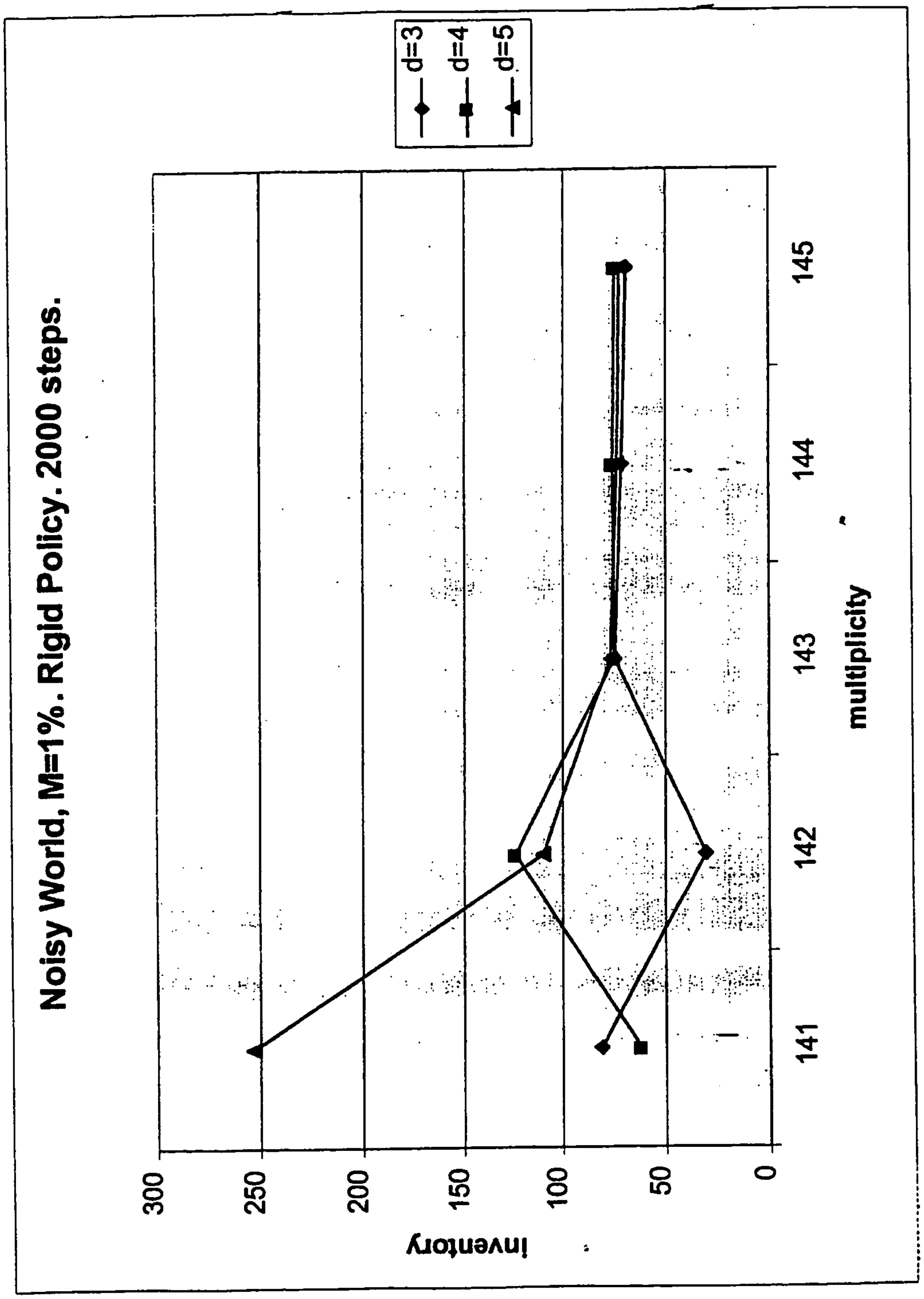

Noisy World, M=1%. Rigid Policy. 2000 steps.
d=3
d=4
d=5
300
250
200
150
100
50
0
Inventory
141
142
143
144
145
multiplicity

FIG. 29

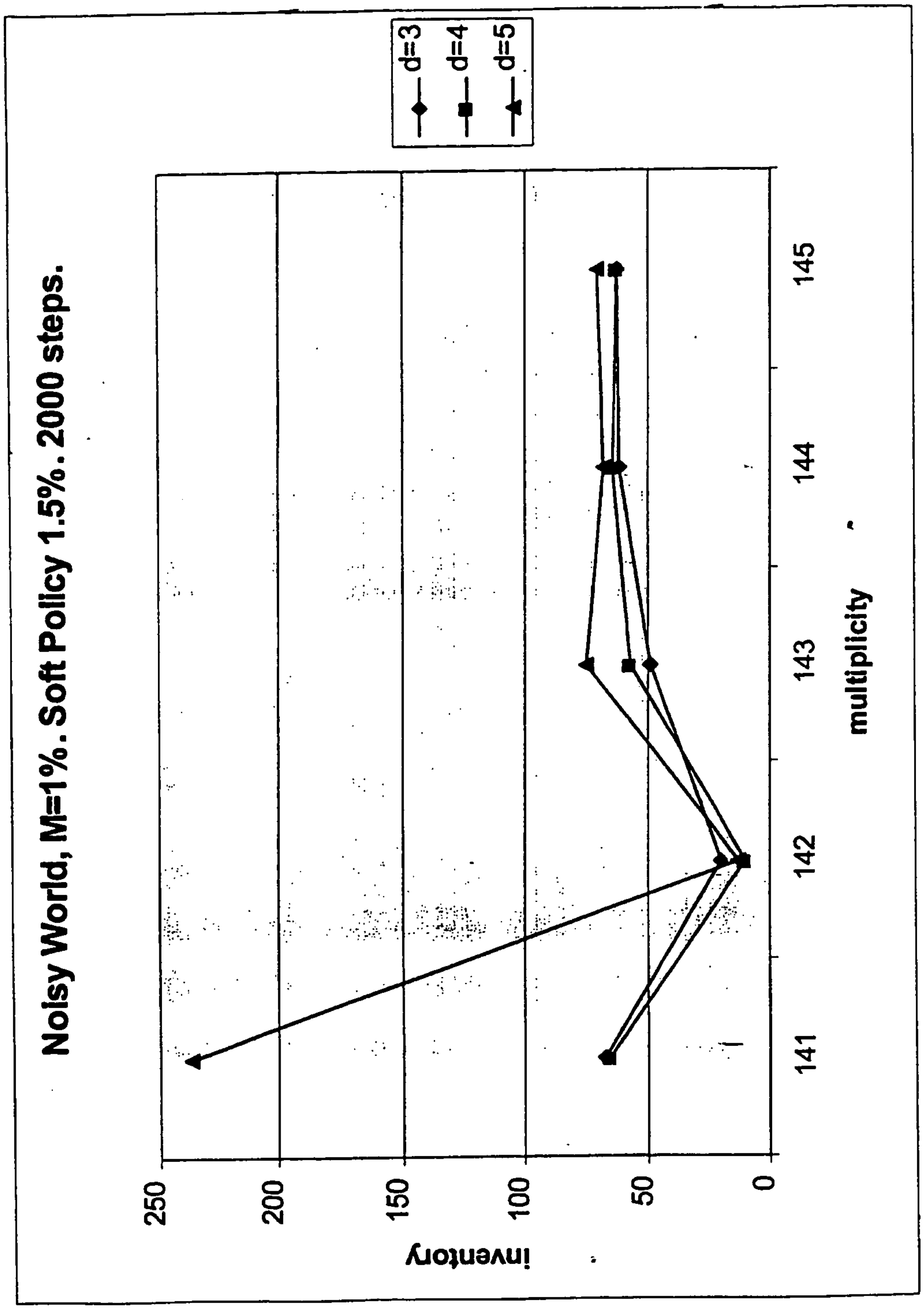

Noisy World, M=1%. Soft Policy 1.5%. 2000 steps.
inventory
250
200
150
100
50
0
141
142
143
144
145
multiplicity
d=3
d=4
d=5

FIG. 30

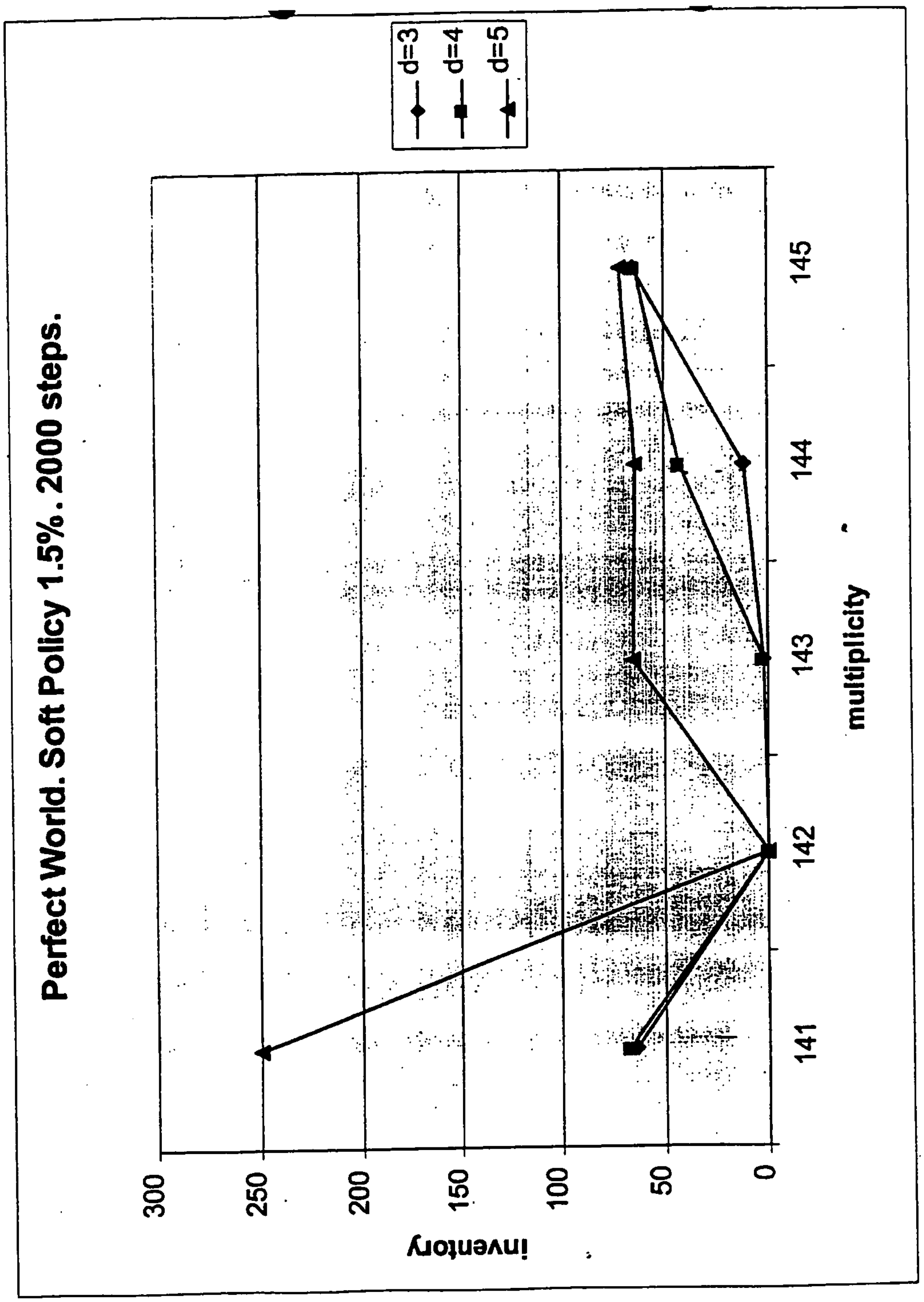

Perfect World. Soft Policy 1.5%. 2000 steps.
d=3
d=4
d=5
inventory
300
250
200
150
100
50
0
141
142
143
144
145
multiplicity

FIG. 31

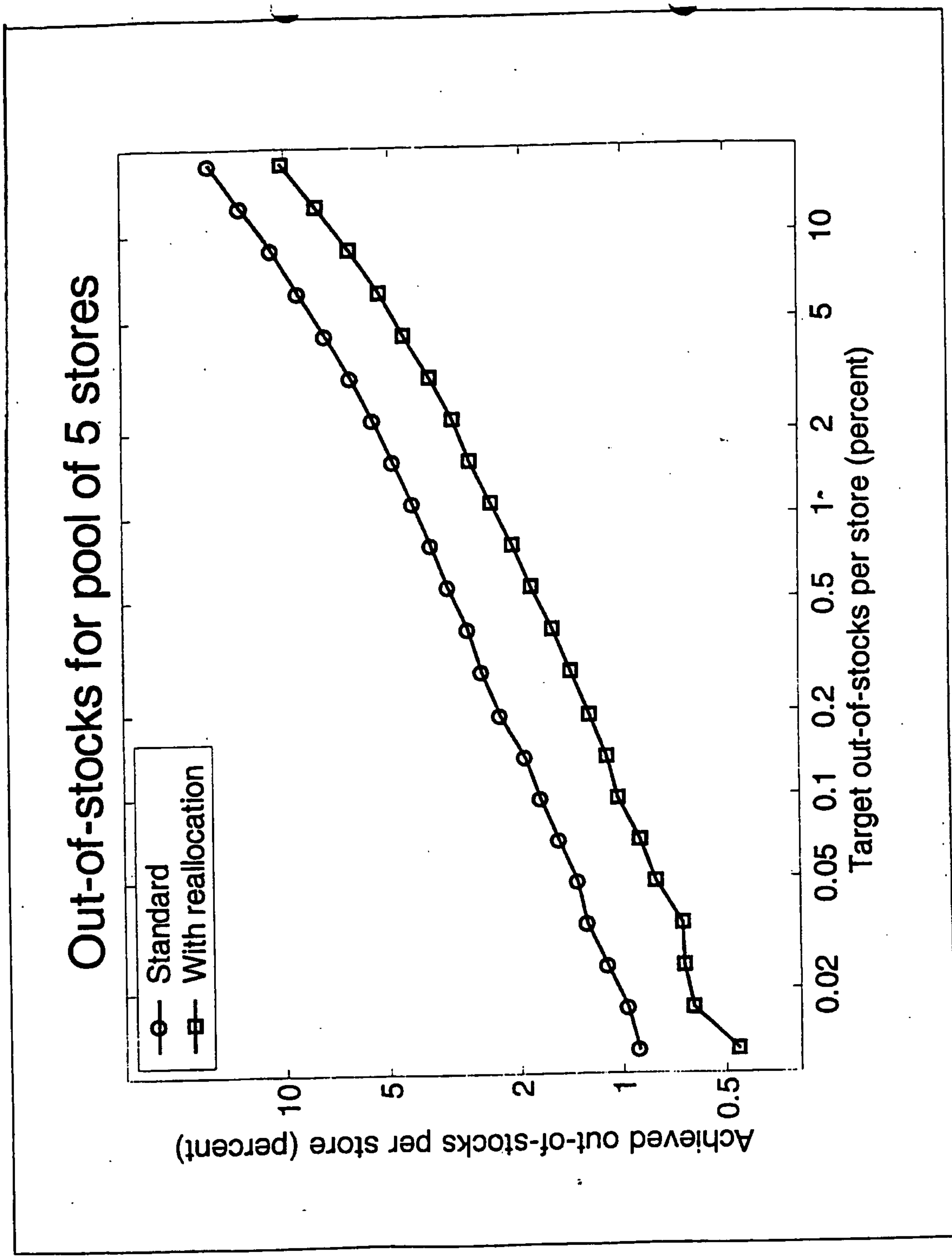

Out-of-stocks for pool of 5 stores
Standard
With reallocation
Target out-of-stocks per store (percent)
Achieved out-of-stocks per store (percent)
0.02
0.05
0.1
0.2
0.5
1
2
5
10
0.5
1
2
5
10

FIG. 32

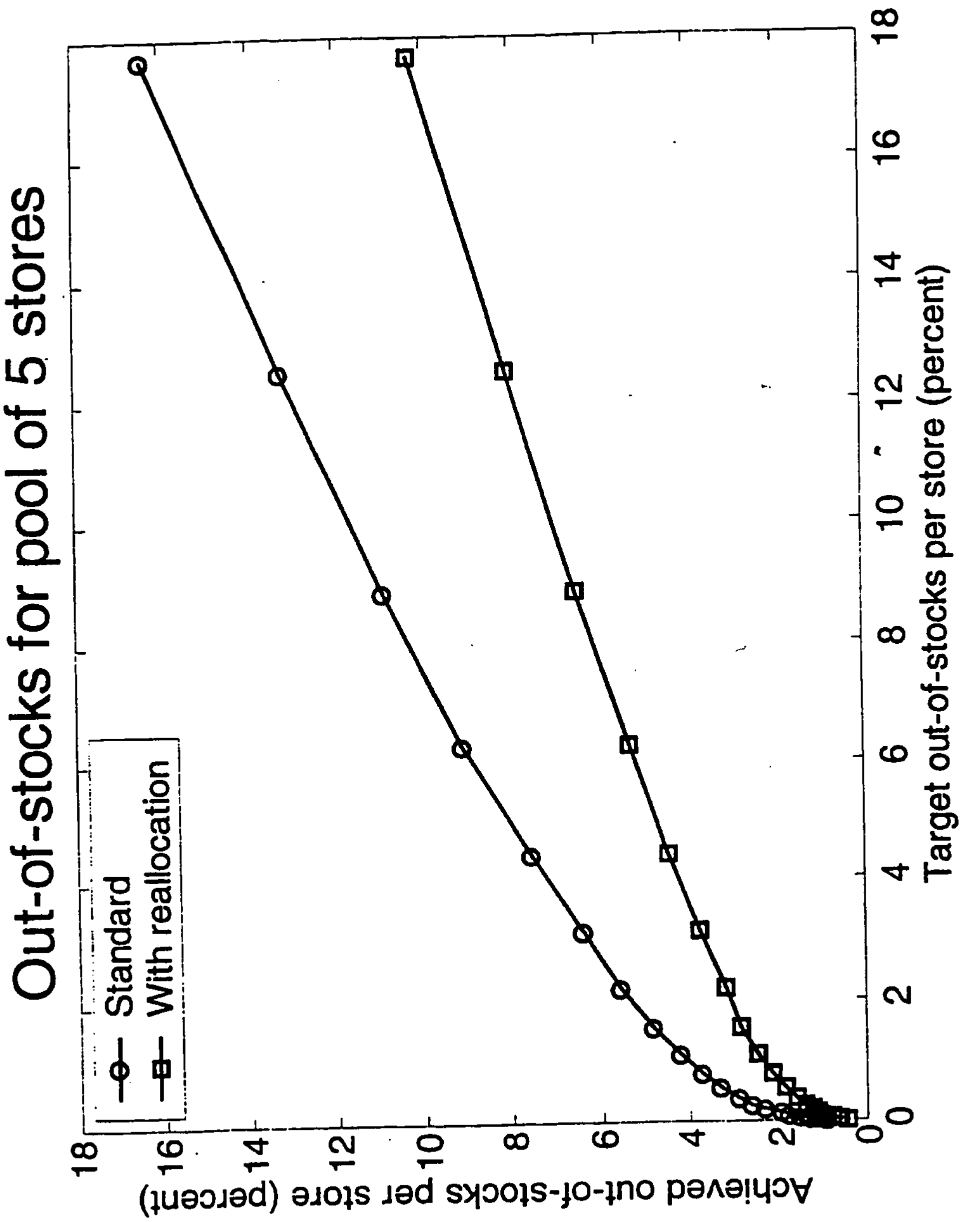

Out-of-stocks for pool of 5 stores
Standard
With reallocation
Achieved out-of-stocks per store (percent)
Target out-of-stocks per store (percent)
0
2
4
6
8
10
12
14
16
18

FIG. 33

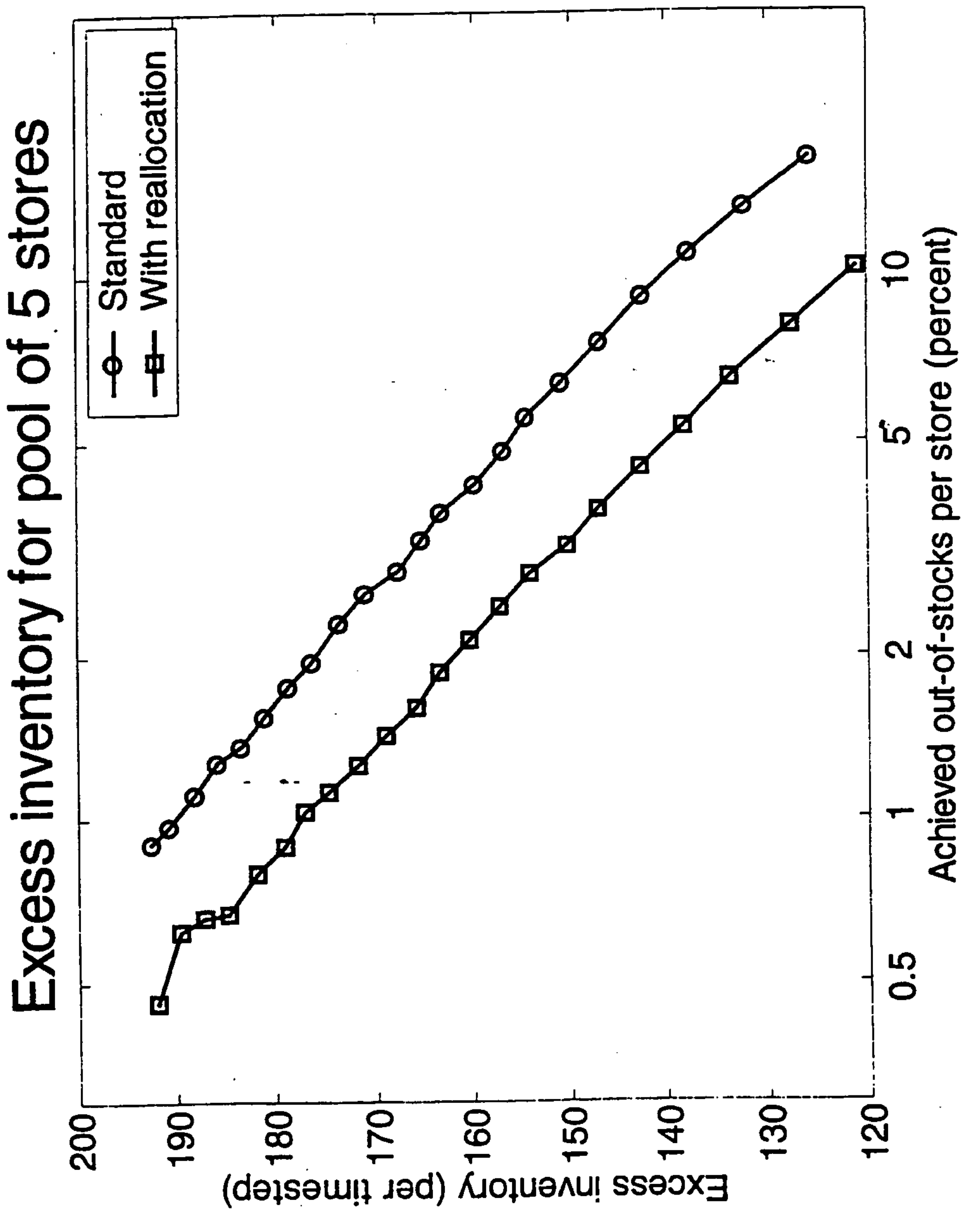

Excess inventory for pool of 5 stores
Standard
With reallocation
Excess inventory (per timestep)
200
190
180
170
160
150
140
130
120
Achieved out-of-stocks per store (percent)
0.5
1
2
5
10

FIG. 34

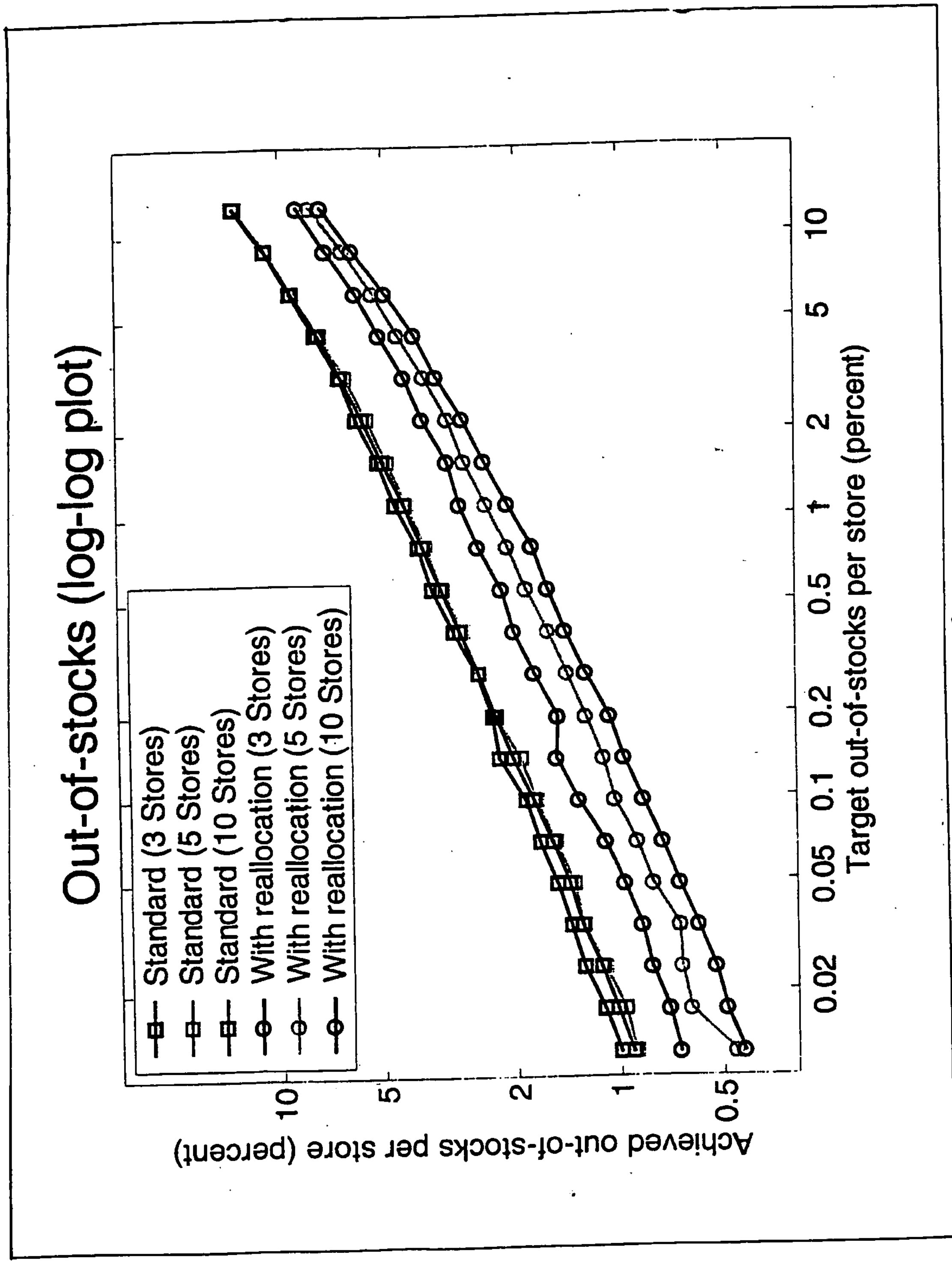

Out-of-stocks (log-log plot)
Standard (3 Stores)
Standard (5 Stores)
Standard (10 Stores)
With reallocation (3 Stores)
With reallocation (5 Stores)
With reallocation (10 Stores)
Achieved out-of-stocks per store (percent)
0.5
1
2
5
10
Target out-of-stocks per store (percent)
0.02
0.05
0.1
0.2
0.5
1
2
5
10

FIG. 36

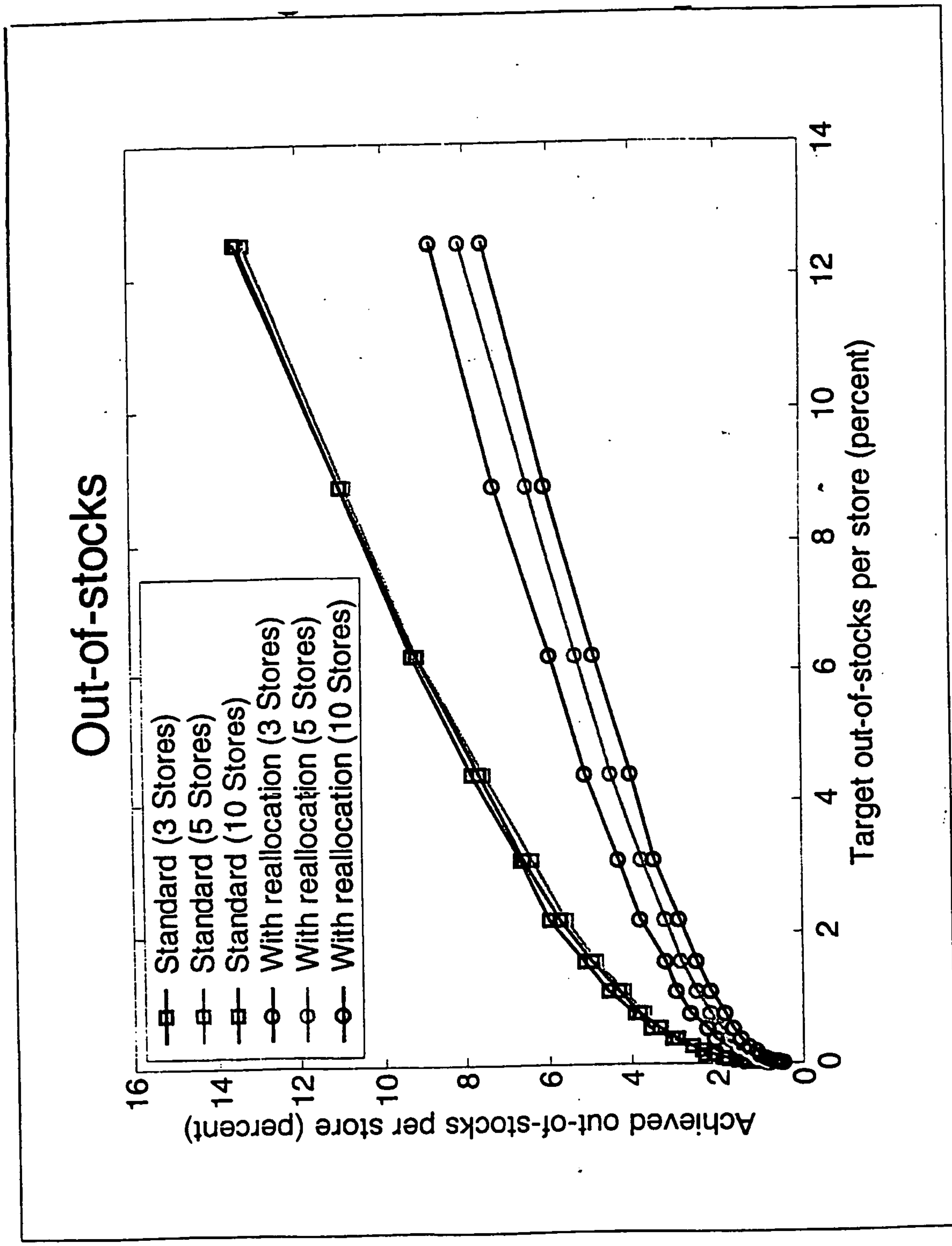

Out-of-stocks
Standard (3 Stores)
Standard (5 Stores)
Standard (10 Stores)
With reallocation (3 Stores)
With reallocation (5 Stores)
With reallocation (10 Stores)
Achieved out-of-stocks per store (percent)
Target out-of-stocks per store (percent)
0
2
4
6
8
10
12
14
16

FIG. 37

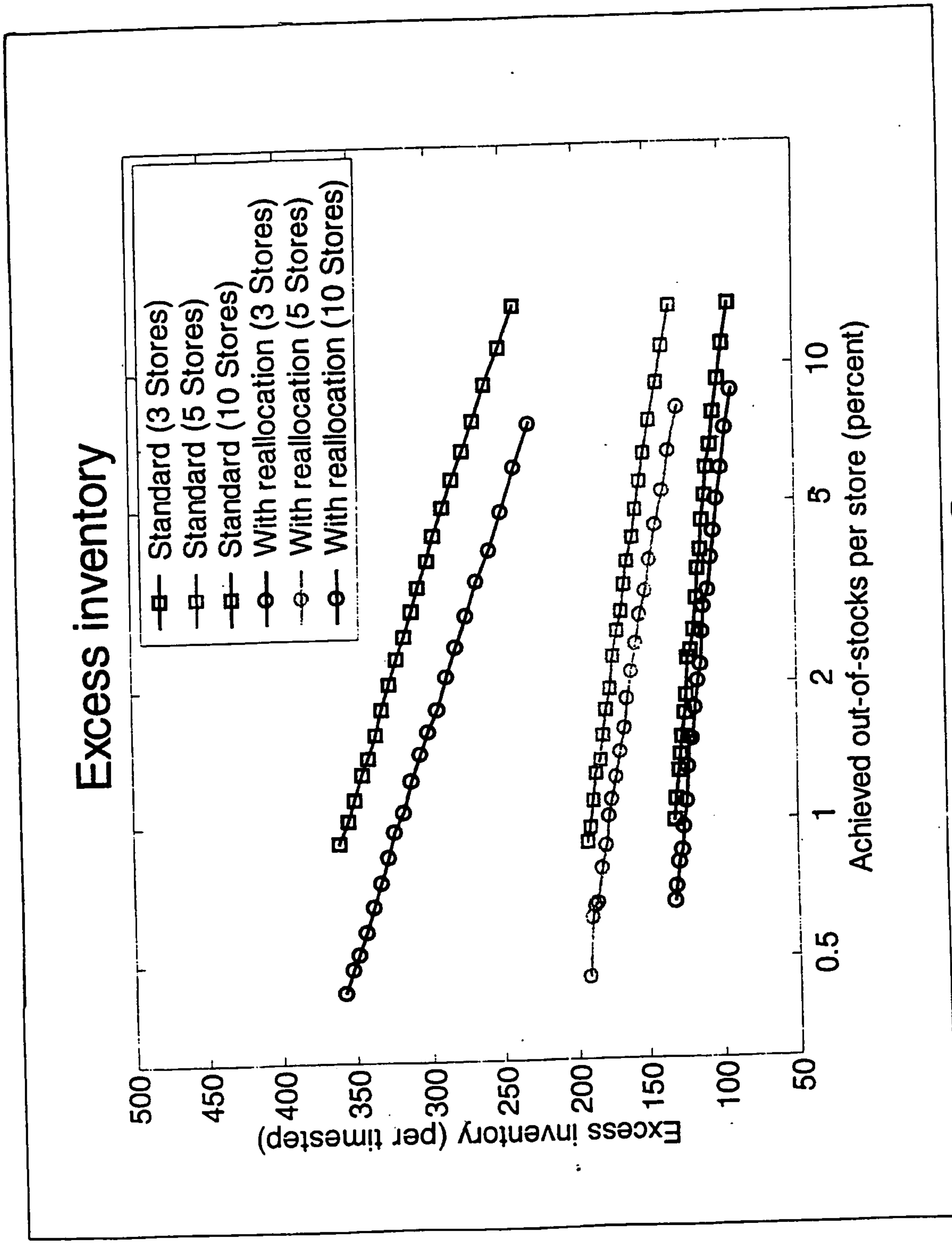

Excess inventory
Standard (3 Stores)
Standard (5 Stores)
Standard (10 Stores)
With reallocation (3 Stores)
With reallocation (5 Stores)
With reallocation (10 Stores)
Excess inventory (per timestep)
500
450
400
350
300
250
200
150
100
50
0.5
1
2
5
10
Achieved out-of-stocks per store (percent)

FIG. 38

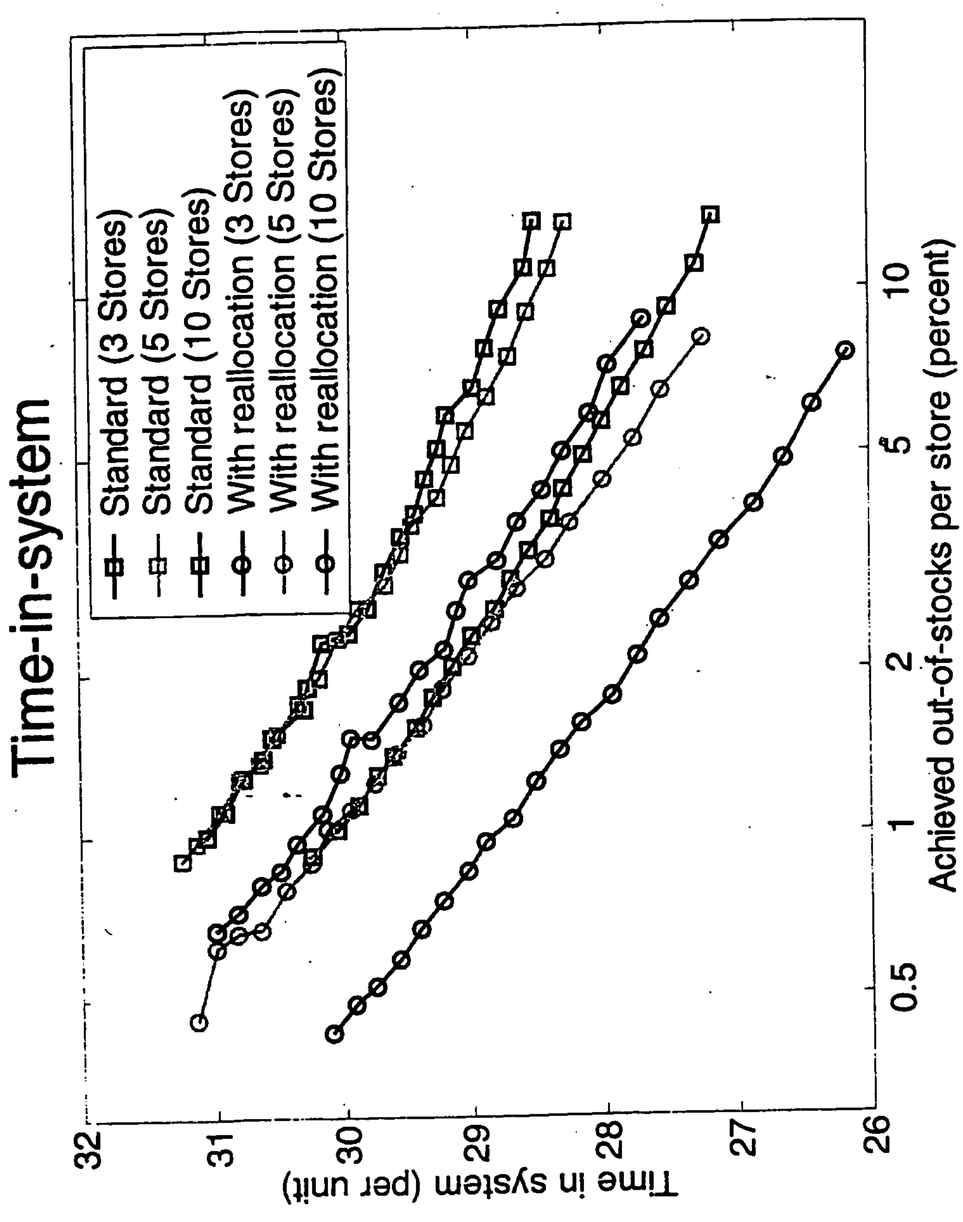

Time-in-system
Standard (3 Stores)
Standard (5 Stores)
Standard (10 Stores)
With reallocation (3 Stores)
With reallocation (5 Stores)
With reallocation (10 Stores)
Time in system (per unit)
32
31
30
29
28
27
26
Achieved out-of-stocks per store (percent)
0.5
1
2
5
10

FIG. 39

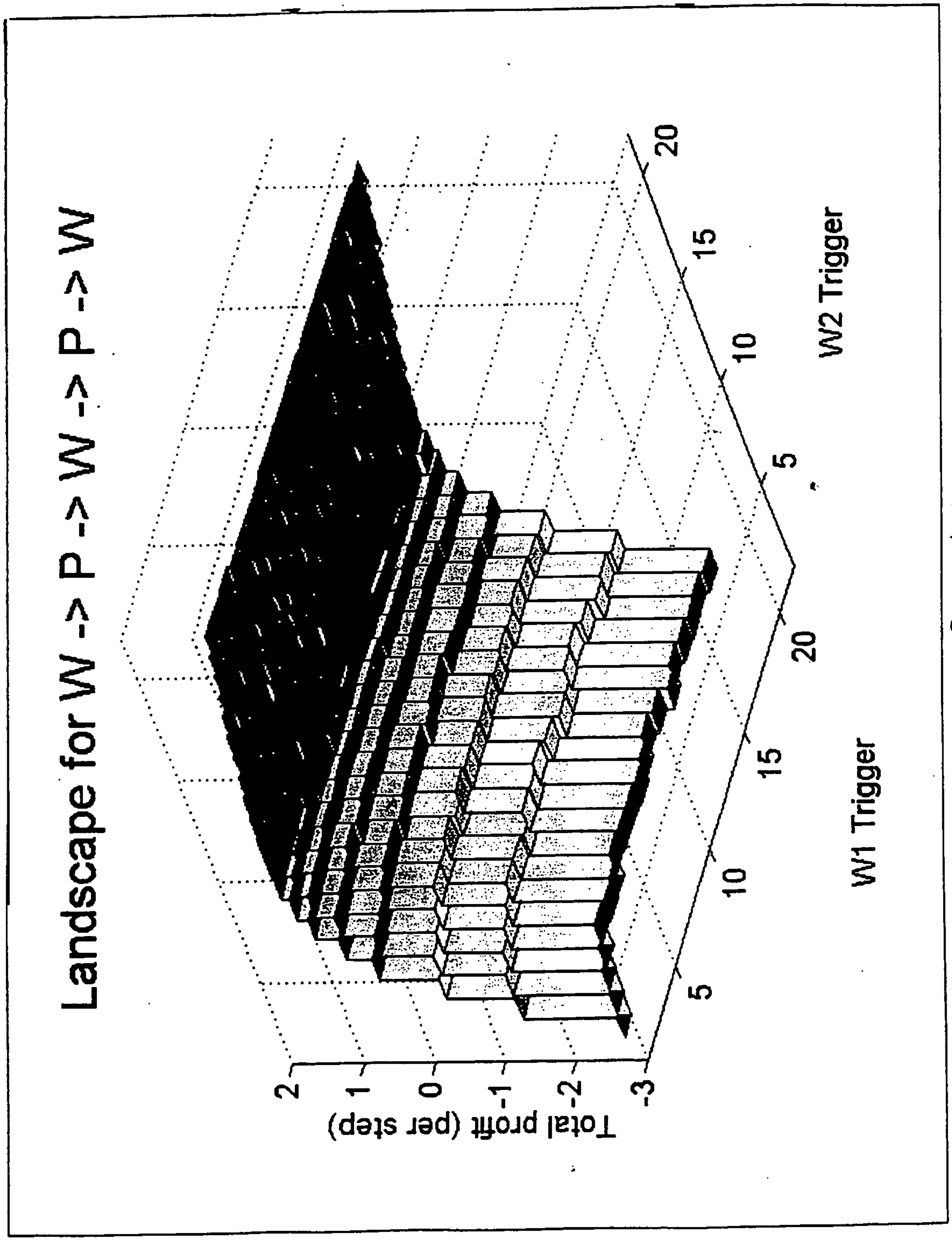

Landscape for W -> P -> W -> P -> W
Total profit (per step)
2
1
0
-1
-2
-3
5
10
15
20
W1 Trigger
5
10
15
20
W2 Trigger

FIG. 40

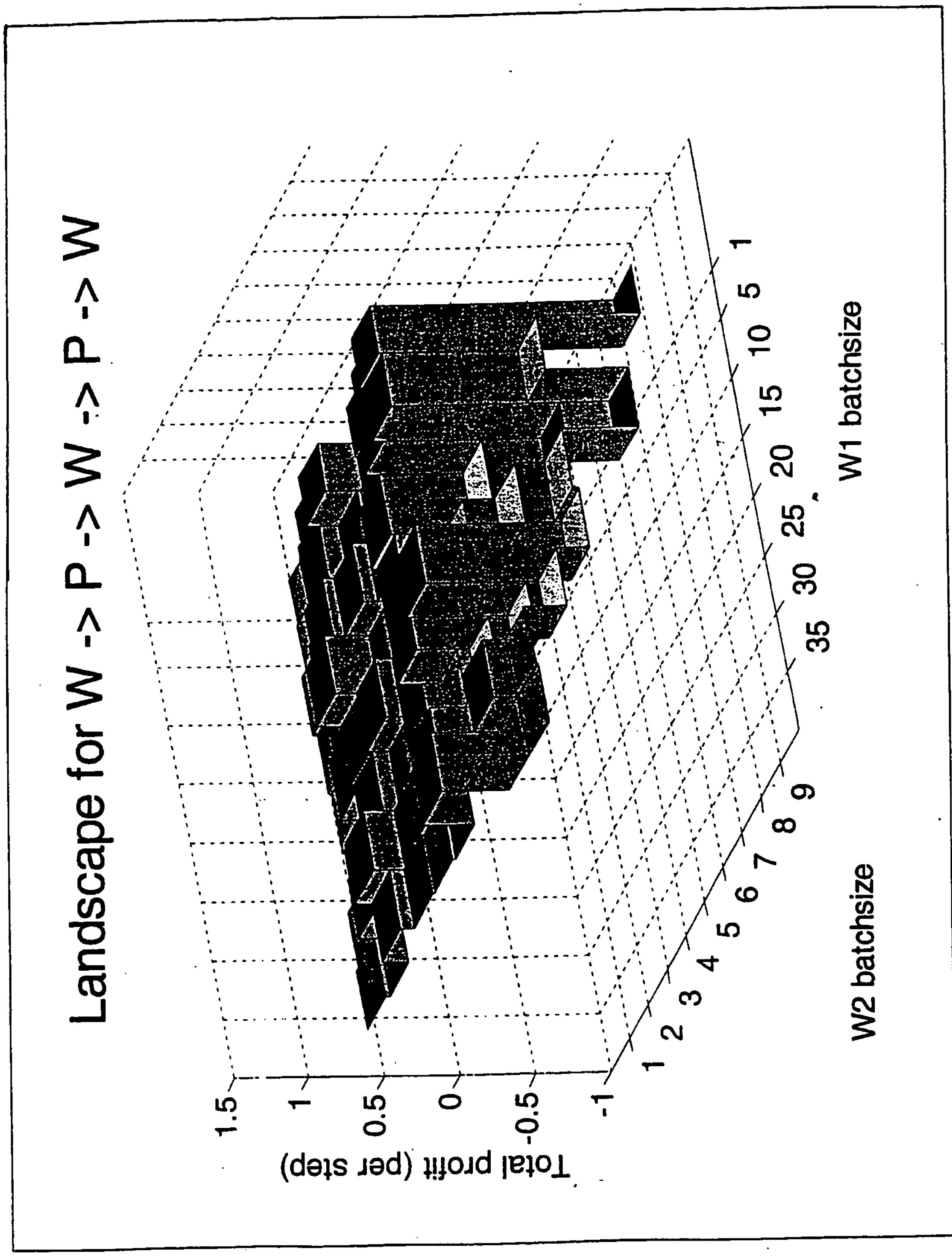

Landscape for W -> P -> W -> P -> W
Total profit (per step)
1.5
1
0.5
0
-0.5
-1
W2 batchsize
1
2
3
4
5
6
7
8
9
W1 batchsize
1
5
10
15
20
25
30
35

FIG. 41

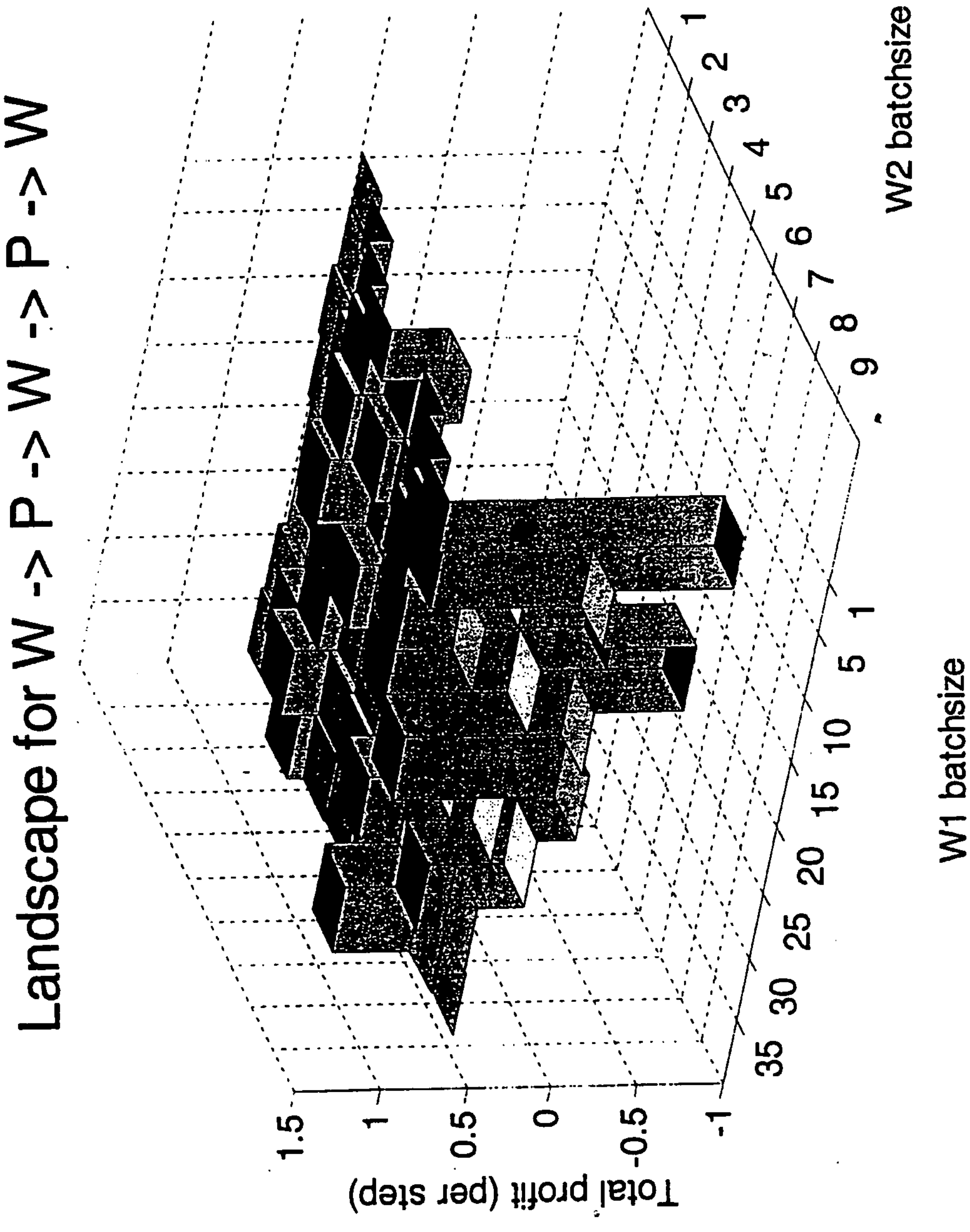

Landscape for W -> P -> W -> P -> W
Total profit (per step)
1.5
1
0.5
0
-0.5
-1
W1 batchsize
1
5
10
15
20
25
30
35
W2 batchsize
1
2
3
4
5
6
7
8
9

FIG. 42

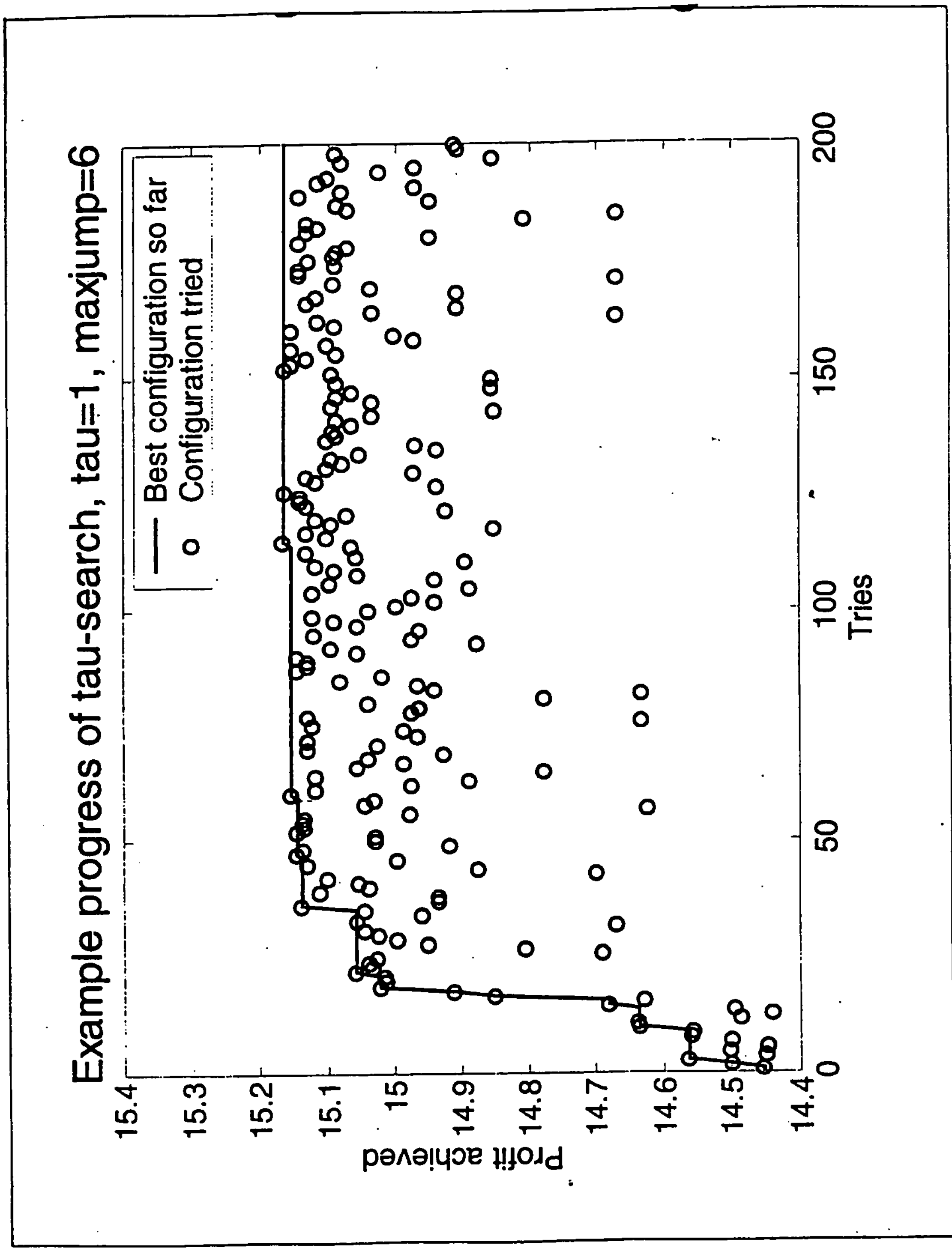

Example progress of tau-search, tau=1, maxjump=6
Best configuration so far
Configuration tried
Profit achieved
Tries
15.4
15.3
15.2
15.1
15
14.9
14.8
14.7
14.6
14.5
14.4
0
50
100
150
200

FIG. 43

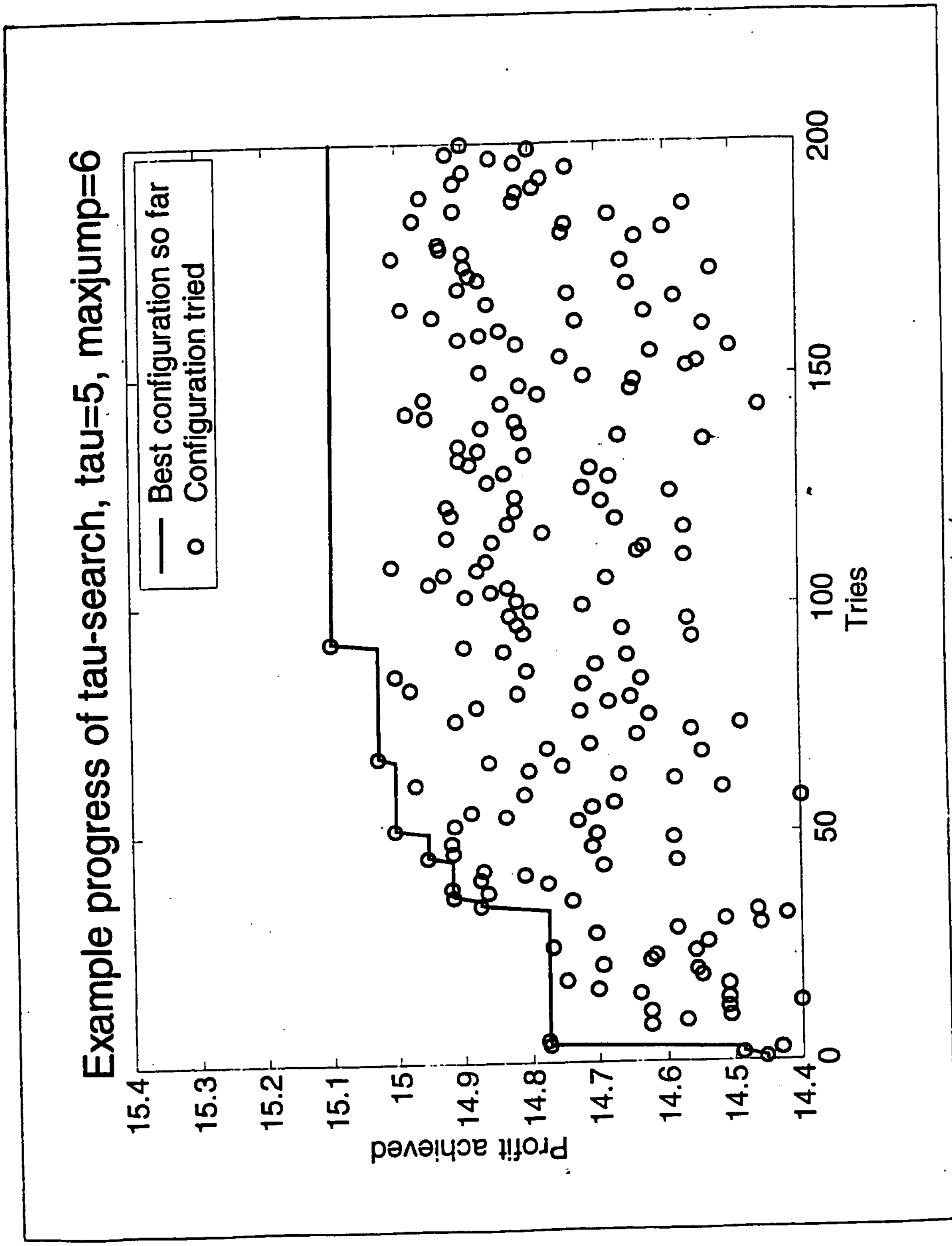

Example progress of tau-search, tau=5, maxjump=6
Best configuration so far
Configuration tried
Profit achieved
Tries
15.4
15.3
15.2
15.1
15
14.9
14.8
14.7
14.6
14.5
14.4
0
50
100
150
200

FIG. 44

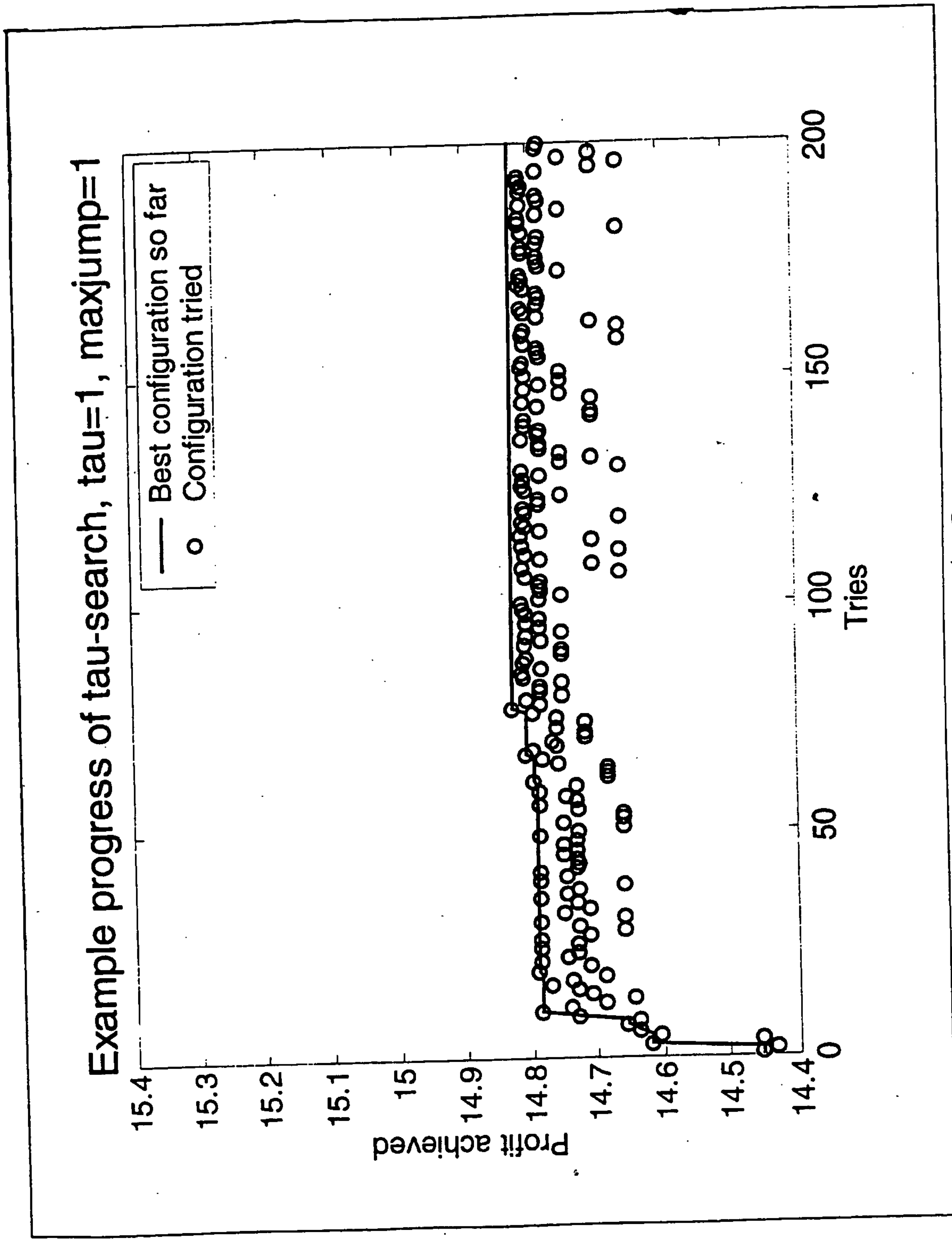

Example progress of tau-search, tau=1, maxjump=1
Best configuration so far
Configuration tried
Profit achieved
Tries
15.4
15.3
15.2
15.1
15
14.9
14.8
14.7
14.6
14.5
14.4
0
50
100
150
200

FIG. 45

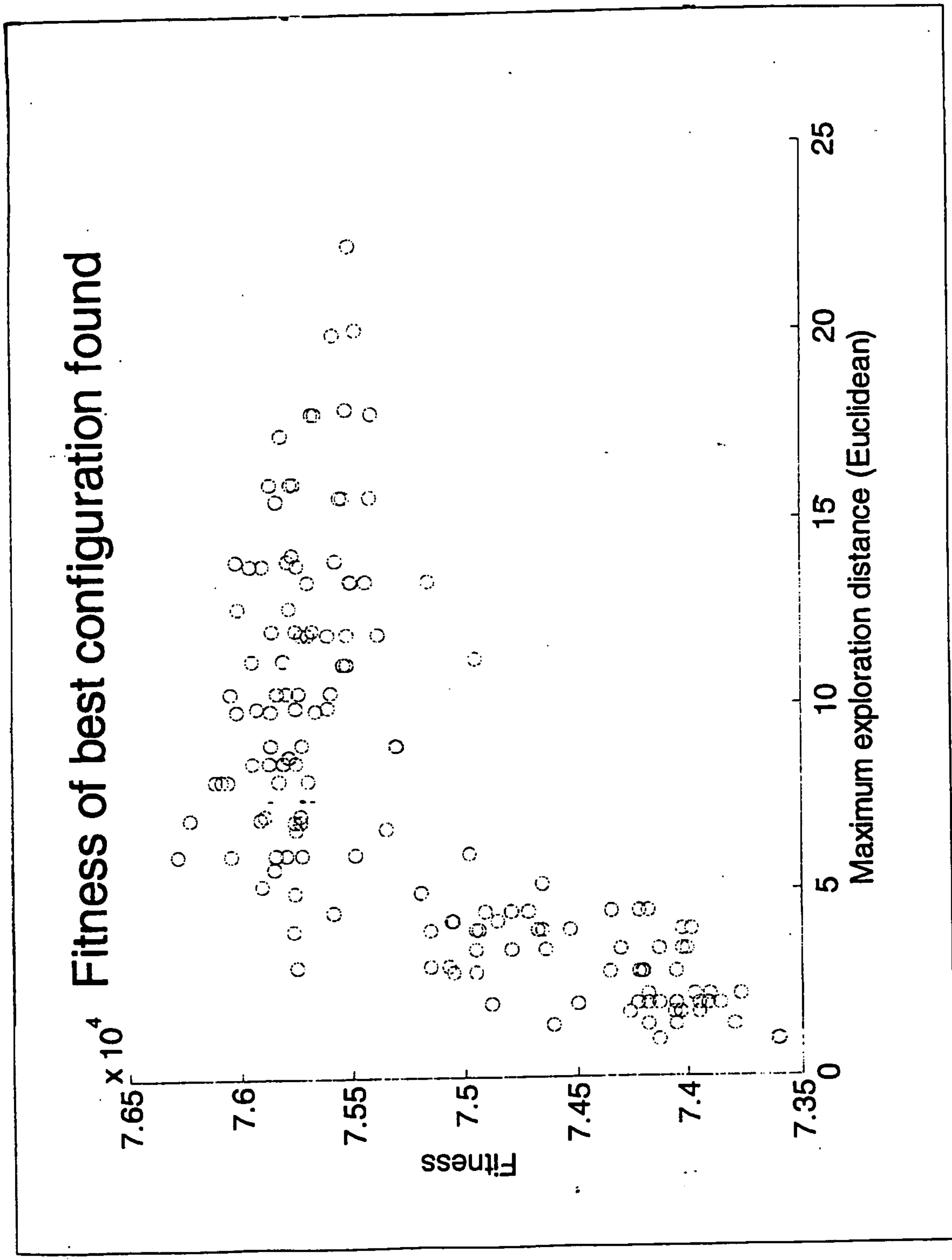

Fitness of best configuration found
x 10^4
7.65
7.6
7.55
7.5
7.45
7.4
7.35
Fitness
0
5
10
15
20
25
Maximum exploration distance (Euclidean)

FIG. 46

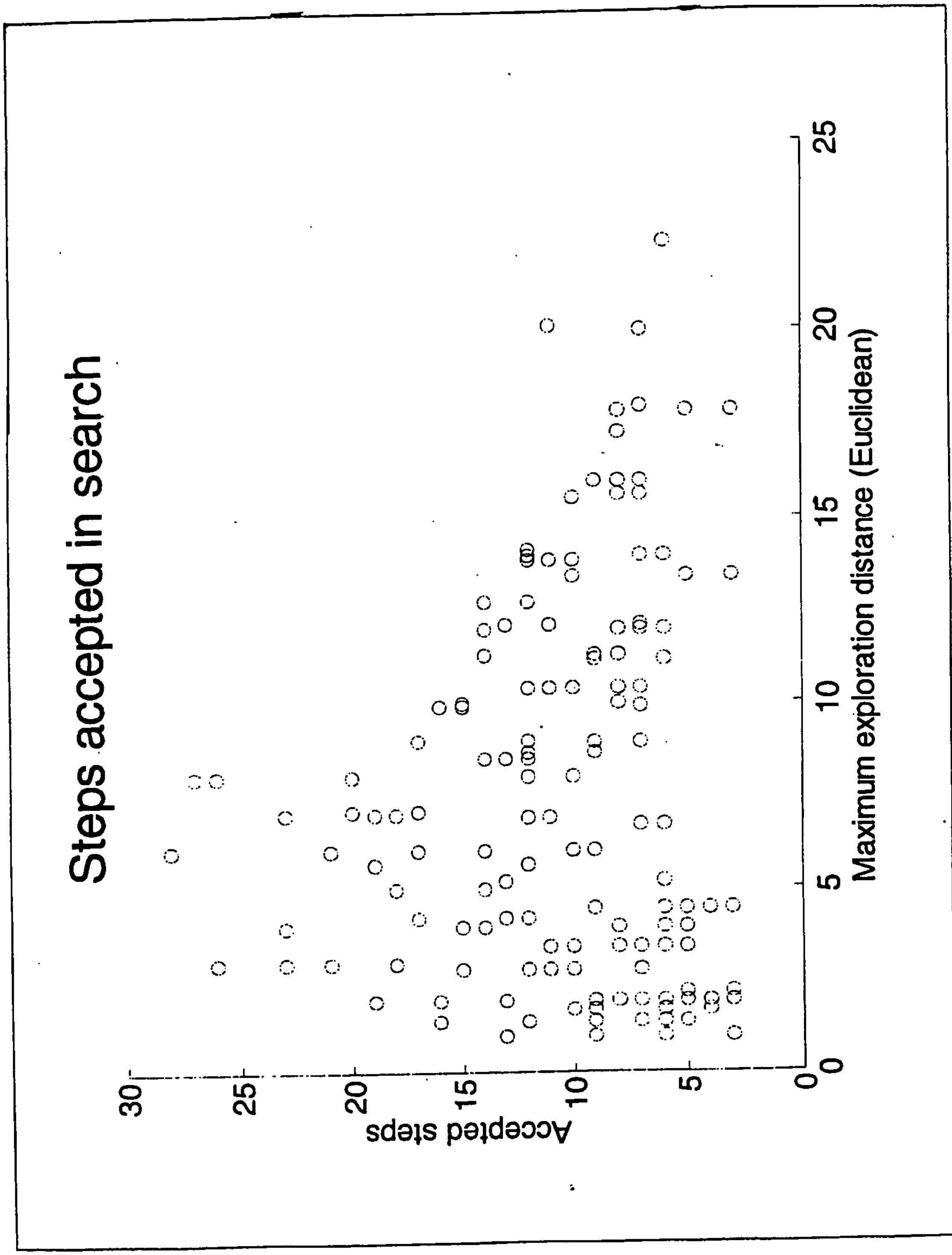

Steps accepted in search
Accepted steps
0
5
10
15
20
25
30
Maximum exploration distance (Euclidean)
0
5
10
15
20
25

FIG. 47

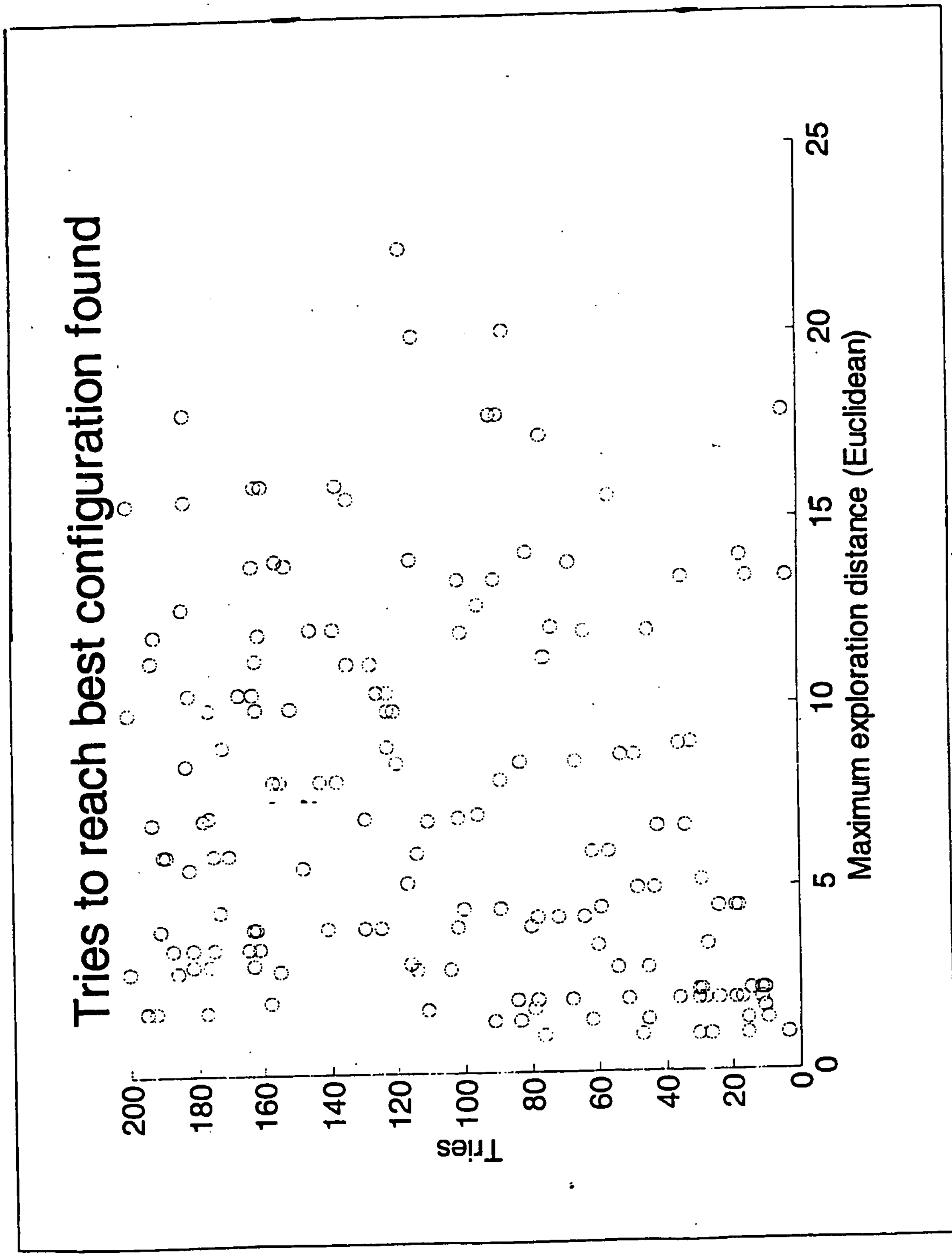

Tries to reach best configuration found
Tries
Maximum exploration distance (Euclidean)
0
20
40
60
80
100
120
140
160
180
200
0
5
10
15
20
25

FIG. 48

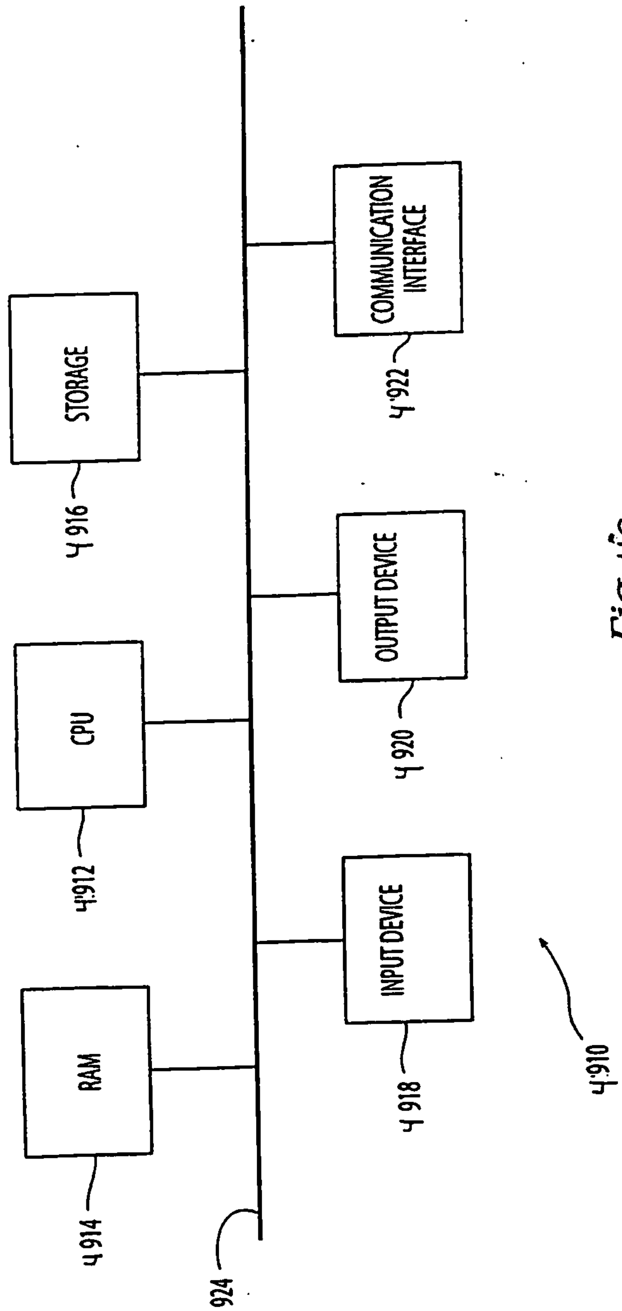

RAM
CPU
STORAGE
INPUT DEVICE
OUTPUT DEVICE
COMMUNICATION INTERFACE
4914
4912
4916
4918
4920
4922
924
4910

Fig. 49

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/21281

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 17/60, 153/00, 19/00.
US CL :705/10, 1, 7, 8, 22, 28, 29
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 705/10, 1, 7, 8, 22, 28, 29

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Dialog, seventeen databases on business and industries.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	US 5,953,707 A [HUANG et al.] 14 September 1999, col.1, line 50-col.2, line 52, col.4, line 45-col.5, line 56, col. 6, line 50-col. 8, line 55, col. 12, line 51-col. 14, line 19, col. 18, line 7-col. 20, line 49, col. 24, line 28-col. 29, line 51, col. 31, line 29-col. 32, line 61, col. 33, line 30-col. 34, line 44, col. 36, line 1-col. 39, line 28, col. 40, line 35-col. 56, line 45, col. 69. line 1-col. 82, line 32, col. 86, line 43-col. 90, line 19, col. 91, line 1-col. 95, line 51, col. 97, line 31-col. 98, line 64, col. 101, line 44-col. 104, line 57, col. 108, line 21-col. 110, line 34.	1-69

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 SEPTEMBER 2000	18 OCT 2000

Name and mailing address of the ISA/US	Authorized officer
Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231	Tod Swann Rugenia Logan
Facsimile No. (703) 308-1396	Telephone No. (703) 308-7791

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/21281

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	US 5,963,919 A [BRINKLEY et al.] 05 October 1999, col.2, lines 35-60, col. 3, lines 25-61, col. 4, line 35-col. 7, line 50, col. 7, line 60-col. 10, line 37, col. 10, line 59-col. 11, line 46, col. 12, line 44-col. 14, line 21	1-69
X,P	US 5,971,585 A [DANGAT et al.] 26 October 1999, col. 4, line 55-col. 7, line 30, col. 8, line 13-col. 9, line 67, col. 10, line 53 - col. 11, line 63, col. 12, line 54-col. 13, line 34, col. 14, line 33-col. 17, line 51, col. 19, line 11-col. 21, line 20, col. 23, lines 4-33	1-69
X,P	US 5,946,662 A [ETTL et al.] 31 August 1999, col. 1, line 56-col. 3, line 10, col. 5, line 7-col. 6, line col. 12, line 44, col. 22, line 24-col. 24, line 5, col. 24, line 46-col. 26, line 58, col. 28, line 1-col. 29, line 47, col. 32, lines 46-65	1-16, 20-38, 49-53, 68, 69
X,P	US 6,078,900 A [ETTL et al.] 20 June 2000, col 3, lines 19-35, col. 3, line 56-col. 6, line 50, col. 8, line 27-col. 11, line 54, col. 13, line 64-col. 14, line 22, col. 15, lines 8-43	1-7, 20, 21, 6-37, 68, 69
A	US 5,930,762 A [MASCH] 27 July 1999, col. 2, lines 15-31, col. 13, line 11-col. 14, line 21, col. 20, line 8-col. 21, line 14, col. 23, line 16-col. 25, line 22, col. 26, line 20-col. 27, line 65, col. 29, line 51-col. 31, line 66, col. 36, line 30-col. 39, line 59, col. 41, line 44-col. 42, line 11	1-6,14, 17, 28-31, 35, 42, 47